PARTS OF THE ORIGINAL DOCUMENT

ARE OF POOR REPRODUCTIVE QUALITY

CERTAINES PARTIES DU DOCUMENT ORIGINAL

SONT DE PIÈTRE QUALITÉ DE REPRODUCTION

Bibliography:

Computer Programs for Creating a Bibliography With Accompanying Kwic Index

D. Clay

Marine Fisheries Division
Gulf Fisheries Centre
Department of Fisheries & Oceans
P.O. Box 5030
Moncton, N.B. Canada
E1C 9B6

March 1987

Canadian Manuscript Report of Fisheries and Aquatic Sciences No. 1910

BIBLIOGRAPHY: Computer programs for creating a bibliography with accompanying KWIC index.

D. Clay

Marine Fisheries Division
Gulf Fisheries Centre
Department of Fisheries & Oceans
P. O. Box 5030
Moncton, N. B. Canada
E1C 9B6

March 1987

Canadian Manuscript Report of
Fisheries and Aquatic Sciences
No. 1910

CANADIAN MANUSCRIPT REPORT OF
FISHERIES AND AQUATIC SCIENCES NO. 1910

MARCH 1987

BIBLIOGRAPHY: Computer programs for creating a bibliography with accompanying KWIC index

by:

D. Clay
Marine Fisheries Division
Gulf Fisheries Centre
Department of Fisheries & Oceans
P.O. Box 5030
Moncton, N. B. CANADA
E1C 9B6

(c) Minister of Suply and Services Canada 1987

Cat. No. FS 97-4/1910E ISSN 0706-6473

Correct citation for this publication:

Clay, D. 1987. Bibliography: Computer programs for creating a bibliography with accompanying KWIC index. Can. Man. Rep. Fish. Aquat. Sci. 1910: iii + 37 p.

ABSTRACT

Clay, D. BIBLIOGRAPHY: Computer programs for creating a bibliography with accompanying KWIC index. Can. Man. Rep. Fish. Aquat. Sci. 1910: iii + 37 p.

A set of three programs to input a bibliography, prepare the references for sorting or a KWIC (key word in context) index, and printout the references in these possible formats. The programs are written in FORTRAN IV for 8 and 16 bit microcomputers.

RESUMÉ

Clay, D. BIBLIOGRAPHY: Computer programs for creating a bibliography with accompanying KWIC index. Can. Man. Rep. Fish. Aquat. Sci. 1910 iii + 37 p.

Un ensemble de trois programmes pour l'entrée des références bibliographiques, préparer des références pour triage ou un index MCEC (mots-clefs en contexte) et imprimer les références dans trois formats différents. Les programmes sont écrits en langage FORTRAN IV pour micro-ordinateurs 8 et 16 bits.

INTRODUCTION

During the course of many research projects extensive bibliographies are often collected. With the increased use of personal micro-computers bibliographies can be recorded on 'floppy disks' almost as easily as on paper 'cards'. Using the following computer programs it is possible for a researcher to gather a bibliography as they normally would, and upon the termination of the project produce an alphabetically or chronologically sorted reference list complete with a keyword index. Although these would not necessarily be published - they could be passed on to others and would certainly improve efficiency for fellow workers in the same and related fields.

This package of computer programs is written in FORTRAN IV for 8 and 16 bit microcomputers running under the CP/M and either MS-DOS or PC-DOS operating systems respectively. The languages used are Nevada FORTRAN (version 2.2) for the 8 bit machines and Utah FORTRAN (version 1.0(2)) for the 16 bit machines, the source code for these implementations have to date been interchangeable.

These programs or earlier versions have been used to produce several unpublished as well as published bibliographies (Clay & Clay, MS1979; Clay, 1980; Clay & Kenchington, 1986). The current version of this package is composed of three programs; the first for data input and minor automatic editing to standardize the style of the author(s) names, the second for printing the bibliography in one of three output formats from two input formats, and the third for creating a "KWIC" (Key Word In Context) index and preparing the bibliography for sorting.

This "KWIC" index lists every word that appears in the title and any special words entered after the reference as 'key words'. The KWIC index can be sorted alphabetically once it has been created. These key words are listed one under the other with the first letter starting in a selected column on the page (usually the center), part of the rest of the title is shown on either side of this "key word" providing its "context". To keep the length of this index within reasonable bounds, many "key words" should be suppressed. These include all prepositions, conjunctions, numbers and the like; words to be suppressed are stored on the file <BIB.KEY> and sufficient blanks have been left to permit additional choices depending on the bibliography and the audience.

The files used with this system are are outlined in Table 1.

PROGRAM DESCRIPTIONS

<BIBIN>

A simplified flowchart of the three programs is shown in Figure 1. To initiate this program, type BIBIN with the file <BIBIN.EXE> on the default drive.

This program permits the user to enter individual references for a bibliography (see Appendix I for source listing). The references can also be entered using any word processor and must be edited and corrected with a text editor or word processor in NON document mode. The advantage of using this program is the resulting standardization of the format which is necessary for later printing and preparing of the KWIC index. The program does some automatic editing of the authors names to ensure a uniform style. This was found necessary due to the great number of persons who often assist in the entering of such bibliographies. At the same stage curly brackets " { } " are added around the special key words entered after the title.

Version 3.0 of this program considers data as four types; the year - four characters in the form "19xx", the author or authors - each entered on a separate line in 24 or less characters, the title and journal credits - in one or more lines of 50 characters or less, and the key words - one or more lines of 48 characters or less.

The year is a four digit integer entered as a string of characters - no checking is done to ensure a legal value. To finish the session type "END" or "end" when prompted for the year.

The author or authors are entered, one per line, with no checking for numeric or other improbable values. To allow sorting by author the surname must be entered first, followed by the Christian names or initials. The legal length is indicated by the symbols " ( ) " at the terminal prompt. Automatic style editing is carried out by removing all blanks after the family name, ensuring that there is a comma between subsequent authors names, and that there is no comma after the last or only author. To pass on to title mode a <CR> is hit for the next author prompt.

The title and journal credits are entered as one or more lines of 50 characters or less. The period "." is used as the end of title indicator before the start of the journal or publisher credits. Thus any periods contained within a title must be replaced by another punctuation mark; a colon, has been found to be satisfactory in most cases where an end of sentence is indicated. When abbreviations contain a period ie. "Gulf of St. Lawrence" the period can usually be omitted. The journal credit can either be entered in a continuous form after the reference or by the users convention could always start on a new title line. The legal length is indicated by the symbols " ( ) " at the terminal prompt. To pass on to key word mode a <CR> is hit for the next title prompt.

Key words are entered in the same manner as titles, except the maximum length of a line is 48 characters. The key words should be separated from each other by blanks, the program will add the curly brackets "{ }" to identify the key words from the title and journal credits. If a leading blank is inserted by mistake it will be removed. Any number of key word lines are permitted thus enabling unique conventions to be developed, such as individual key words for different purposes on separate lines. The legal length is indicated by the symbols " ( ) " at the terminal prompt. To pass on from key word mode a <CR> is hit for the next prompt.

The bibliography is stored on a computer file, this can be either a new file or an existing one. It is an advantage to have each entry session on a unique disk file, then after a visual edit and check with a text editor the 'corrected' file can be concatinated to the master file. To facilitate this process, when one file is used for many sessions, at the end of each session an 'end of file' flag is written in the form:

9999 zzzzzzzzzz end of file.

this flag is not erased by the next session and must be deleted with an editor. This enables checking to be done a session at a time and leaves flags to indicate progress. The bibliography is stored as an ASCII file - therefore it is imperative that NON-document mode is used with any text editors or word processors for editing.

In the storage files, two blanks have been left between the year and the author to permit the addition of an alphanumeric code. In this version no use is made of this space.

<BIBPRT>

To initiate this program, type BIBPRT with the file <BIBPRT.EXE> on the default drive.

This program prints the bibliography in three output formats (see Appendix II for source listing). These vary in width, two of them are in bibliographical form with reference numbers added sequentially, the third is the original storage or master format. The first two are 50 and 86 characters wide respectively and have been designed for two column reports, such as this, and full page reports (Figure 2). The master format, 80 columns wide, is used after sorting to return the bibliography to its original form, it can also be used to make an alternate 'master' listing with the added key words deleted. The input can be in two different formats, the most common is the master format, the other is the sorting format with its leading alpha-numeric identifier. After sorting the data can be printed directly from this latter format and then erased without re-formatting to the original form. This might be done in a case where the master file is in alphabetical order and a second printing is required in chronological order.

The parameter file <BIB.PAR> must be on the default drive if the printing is to be done from the sorting format. This file is created by the program <BIBFMT> when the sorting file is made and used by the program <BIBPRT> to unscramble the references in sorting format.

If large bibliographies are being constructed and a hard disk is not available then multi-disk files are permitted under MS-DOS or PC-DOS. This can also be useful if an interim combined printout is required of several small files, possibly all in different stages of editing and correcting.

<BIBFMT>

To initiate this program, type BIBFMT with the file <BIBFMT.EXE> on the default drive.

This program re-formats the bibliography in three different ways (see Appendix III for source listing). All transformations are carried out on

the master format. The first prepares the bibliography for sorting, the second creates the KWIC index (Figure 3) and the third creates a string format reference from the original multi-line master format. The string format contains one or two lines of up to 248 characters on each. The date, author, title and keywords occur in an unformatted endless string, these are available for use by other programs that users may write, eg. to reformat the reference title to something other than 50 character width.

Two parameter files are required on the default drive, these are <BIB.PAR> and <BIB.KEY>. As with <BIBPRT> multi-disk files are permitted, although in this case the KWIC index file is generally considerably larger than the original bibliography, often three or more times larger. This creates a difficult situation for over sized files when no hard disk is available.

The program prepares for sorting of the bibliography by creating, for every line in the file, an alpha-numeric identifier composed from the author(s) name, the date, and the line number within the reference. This identifier can be stripped off by one of the options in <BIBPRT> to return to the master format.

The KWIC index is generated by creating an endless string format and printing a portion of it in a file. The user selects the KWIC file width (<150 characters) and the 'key word column' - usually the center of the page). A normal page is about 70 columns wide (pica type) with the center at about 40. Each subsequent word in the reference is then placed in this column and the adjacent words in the title displayed within the 'window' of the file width. The reference number is placed in the first four columns. The index can be sorted using the 'key word column' as the sort field.

The string format is the intermediate form used to build the KWIC index.

CONVENTIONS

The programs have, for convenience, been designed with certain conventions in mind. These could be changed by anyone familiar with FORTRAN programming. These conventions are:-

1) the bibliography is either sorted alphabetically by authors surname or chronologically;
2) papers by the same author are in order of date, the earliest first;
3) papers by more than one author follow those by the senior author alone;
4) all authors, including junior authors, are listed surname first to facilitate sorting; and
5) as mentioned above, periods located within a title must be eliminated or replaced with another punctuatiuon mark if a KWIC index is to be prepared.

Many other conventions can be adopted to facilitate data entry, editing, and style, these need not be a function of the programs. The date preceding the author should be the year of publication. If need be the date of a meeting, etc can often be incorporated into the journal credits. The authors Christian name(s) can all be represented by initials to simplify sorting - especially where an author sometimes uses initials and sometimes uses their full name.

The title of each item should be reproduced with as few changes from its original form as possible. However, with most micro-computers the character set is limited to those symbols found on a standard North American typewriter, thus there are no accents, non-English letters, italics or underlines. Because of this, we have tended to translate foreign language titles to English and place the translation within square brackets " [ ] " and placed the phrase " (In 'language') " after the journal credits.

Earlier bibliographies have used only the upper case character set as much of the original work was done on mainframe computers with terminals without lower case. The sample bibliography shown here uses upper and lower case. It should be noted that key word suppression becomes more complicated as the list must contain all variants of the word as it may appear. The definite article "the" must be suppressed as "the", "The", and possibly even "THE".

All words in the title are considered key words unless they are suppressed. Therefore to avoid redundancy in the add-on key word list, the user should not include any word already present in the title.

Due to the automatic editing carried out on the authors' names, any surnames composed of two or more words (eg. El Bolock,A.R.) must be

hyphenated to maintain their spacing. After the references have been entered there is no further automatic editing, thus the hyphen could be removed during the proof reading/editing.

Hypens and dashes are handled differently when they occur in the title. If the last non blank character is a '-' it is considered a hypen and will be removed in the string format and in the KWIC index. Any other occurrence of '-' in the title will have spaces placed around it for the KWIC index and the string format.

SORTING THE BIBLIOGRAPHY

The bibliography can be sorted using any commercially available sort utility. If using MS-DOS or PC-DOS the SORT command available with the system on most 16 bit micro computers is adequate for this purpose. As the sort identifier added by <BIBFMT> begins in column 1 the command to sort the bibliography would be:-

```
SORT <BIBLIO.SRT >NEWFILE.SRT
```

where BIBLIO.SRT is the bibliography file with the leading identifier prepared by the program <BIBFMT> and NEWFILE.SRT is the newly sorted output file.

The KWIC index could be sorted as follows:-

```
SORT /+40 <BIBQIK.DAT >NEWQIK.SRT
```

where BIBQIK.DAT is the KWIC index prepared by the program <BIBFMT>, NEWQIK.SRT is the newly sorted output file, and the sort begins in column 40 - the column selected by the user as the key word column.

If using CP/M then another sort utility will be required. We have used SUPERSORT on both 8 and 16 bit machines and found it adequate.

PROGRAM AVAILABILITY

The source code and run time compiled versions of these programs are available from the author when the request is accompanied by a blank formatted double sided floppy disk.

DISCLAIMER

The mention of any commercial product in this report does not constitute an endorsement by either the author or the Department of Fisheries and Oceans.

REFERENCES

Clay,D. and H.Clay. MS1979. Bibliography of the hake family (Pisces: Merluccidae). ICNAF Res. Doc. 79/VI/78. Serial No. 5381.

Clay,D. 1980. Growth and production of the African catfish (Clarias gariepinus). PhD Thesis. University of Reading, Berks, UK. 391 pp.

Clay,D. and T.J.Kenchington. 1986. World bibliography of the redfishes and rockfishes (Sebastinae, Scorpaenidae). Can. Tech. Rep. Fish. Aquat. Sci. 1429:iii +303pp.

TABLE 1. Files used with this bibliographic package and a brief description of the function of each.

File name	File type	File description
<BIBIN.FOR>	source	input program word processor can be substituted for this function
<BIBIN.EXE>	executable	compiled version of <BIBIN.FOR>
<BIBPRT.FOR>	source	printing program
<BIBPRT.EXE>	executable	compiled version of <BIBPRT.FOR>
<BIBFMT.FOR>	source	formatting program for KWIC index
<BIBFMT.EXE>	executable	compiled version of <BIBFMT.FOR>
<BIB.PAR>	parameter	parameters used in sorting the bibliography - required to re-format a sorted file; this file is created by <BIBFMT> and used by <BIBPRT>
<BIB.KEY>	parameter	sample file of key words to be suppressed
<BIBU.KEY>	parameter	sample file of key words to be suppressed in upper case
<BIB.DOC>	documentation	this manuscript report
<BIBLIO.DAT>	default	file for master bibliography
<BIBSRT.DAT>	default	intermediate file prepared for sorting
<BIBQIK.DAT>	default	unsorted KWIC index
<SAMPLE.BIB>	sample	bibliography file
sort utility	commercial	a commercial sort routine - we have used SUPERSORT and SORT
word processor	commercial	a commercial word processor - we have used WORDSTAR, MULTIMATE, and PC-WRITE

FIGURE 1. Simple flowchart of the interactions of the three bibliography programs.

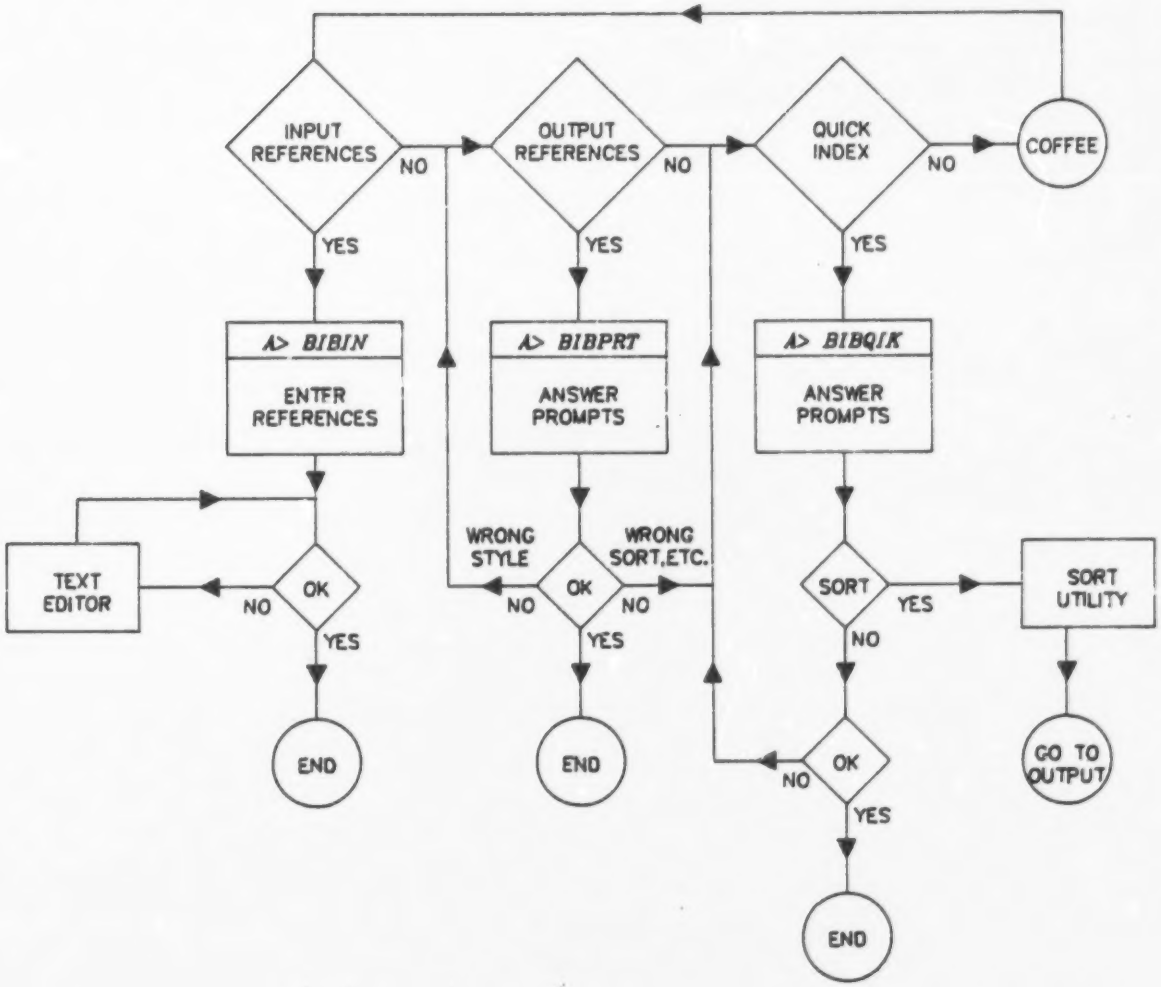

FIGURE 2. Sample bibliography as entered in two separate sessions (not end of file markers). The bibliography is printed out in the two formats, 50 column and 86 column wide.

SAMPLE BIBLIOGRAPHY

1979	Clay,D., Clay,H.	Bibliography of the hake family (Pisces: Merluccilidae). ICNAF Res. Doc. 79/VI/78. Serial No. 5381.
1976	Stearns,S.C.	Life-history tactics: A review of the ideas. Quar. Rev. Biol. 51(1):3-47. (evolution reproduction mortality)
1986	Pieterse,A.J.H., Roos,J.C., Roos,K.I., Piennaar,C.	Preliminary observations on cross-channel and vertical heterogeneity in environmental and algological parameters in the Vaal River at Balkfontein, South Africa. Water SA 12(4):173-184. (temperature TDS turbidity)
1986	Ayers,G.S., Hoopingarner,R.A.	The potential for fixed-land honey production. Amer. Bee J. 126(12):805-808. (forage bees nectar)
1987	Ayers,G.S., Hoopingarner,R.A.	Research imperatives for fixed-land honey production. Amer. Bee J. 127(1):39-41. (forage herbicides)
1986	Clay,D., Kenchington,T.J.	World bibliography of the redfishes and rockfishes (Sebastinae, Scorpaenidae). Can. Tech. Rep. Fish. Aquat. Sci. 1429: iii +303pp.
1984	Dabrowski,K., Bardeça,R.	Mouth size and predicted food size preferences of three cyprid fish species. Aquaculture 40:41-46.
1984	Dabrowski,K.	The feeding of fish larvae: present "state of the art" and perspectives. Reprod. Nutn. Develop. 24(6):807-833. (growth mortality)
1982	Kettleborough,I.	Nevada fortran: Programmer's reference manual. Ellis Computing Inc., San Francisco, USA. 166pp. (compiler computer language 8-bit Z-80 CPM)
1985	Kettleborough,I.D.	Utah fortran: Programmer's reference manual. Ellis Computing Inc., Reno, Nevada, USA. 222pp. (MS-DOS PC-DOS compiler language 16-bit computer)
9999	ZZZZZZZZZZ	END OF FILE
1977	Wells,P.G.	A selected bibliography on rainbow trout, Salmo gairdneri Richardson, with particular reference to studies with aquatic toxicants. Can.Env.Prot. Service, Econ.Tech.Rev.Rept. EPS-3-AR-77-1. 109pp.
1970	Marshall,K.E., Keleher,J.J.	A bibliography of the lake trout, Cristivomer namaycush (Walbaum): 1929-1960. F.R.B.Can.Tech. Rept. 176: iv + 60pp.
1976	Van der Waal,B.C.W.	[An ecological study of the fish of the east Caprivi, with sampling from the Bantu fishery.] Ph.D. Thesis, Rand Africaans Univer. Johannesburg, Republic of South Africa. (In Africaans). (age swamp catfish Clarias)
1972	El-Bolock,A.R.	Use of vertebrae for determination of age and growth of Nile catfish, Clarias lazera. U.A.R. Bull. Inst. Ocean. Fish. 2:53-82.
9999	ZZZZZZZZZZ	END OF FILE.

FIGURE 2. continued

Sorted sample bibliography 86 columns wide.
Keywords removed.

[1]Avers,G.S., Hoopingarner,R.A. (1986)The potential for fixed-land honey production. Amer. Bee J. 126(12):805-808.

[2]Avers,G.S., Hoopingarner,R.A. (1987)Research imperatives for fixed-land honey production. Amer. Bee J. 127(1):39-41.

[3]Clay,D., Clay,H. (1979)Bibliography of the hake family (Pisces: Merlucciidae). ICNAF Res. Doc. 79/VI/78, Serial No. 5381.

[4]Clay,D., Kenchington,T.J. (1986)World bibliography of the redfishes and rockfishes (Sebastinae, Scorpaenidae). Can. Tech. Rep. Fish. Aquat. Sci. 1429: iii +303pp.

[5]Dabrowski,K. (1984)The feeding of fish larvae: present "state of the art" and perspectives. Reprod. Nutn. Develop. 24(6):807-833.

[6]Dabrowski,K., Bardega,R. (1984)Mouth size and predicted food size preferences of three cyprid fish species. Aquaculture 40:41-46.

[7]El-Bolock,A.R. (1972)Use of vertebrae for determination of age and growth of Nile catfish, Clarias lazera. U.A.R. Bull. Inst. Ocean. Fish. 2:53-82.

[8]Kettleborough,I. (1982)Nevada fortran: Programmer s reference manual. Ellis Computing Inc., San Francisco, USA. 166pp.

[9]Kettleborough,I.D. (1985)Utah fortran: Programmer s reference manual. Ellis Computing Inc., Reno, Nevada, USA. 222pp.

[10]Marshall,K.E., Keleher,J.J. (1970)A bibliography of the lake trout, Cristivomer namaycush (Walbaum): 1929-1969. F.R.B.Can.Tech. Rept. 176: iv + 60pp.

[11]Pieterse,A.J.H., Roos,J.C., Roos,P.I., Piennaar,C. (1986)Preliminary observations on cross-channel and vertical heterogeneity in environmental and algological parameters in the Vaal River at Balkfontein, South Africa. Water SA 12(4):173-184.

[12]Stearns,S.C. (1976)Life-history tactics: A review of the ideas. Quar. Rev. Biol. 51(1):3-47.

[13]Van der Waal,B.C.W. (1976)[An ecological study of the fish of the east Caprivi, with sampling from the Bantu fishery.] Ph.D. Thesis, Rand Africaans Univer. Johannesburg, Republic of South Africa. (In Africaans).

[14]Wells,F.G. (1977)A selected bibliography on rainbow trout, Salmo gairdneri Richardson, with particular reference to studies with aquatic toxicants. Can.Env.Prot. Service, Econ.Tech.Rev.Rept. EPS-3-AR-77-1. 109pp.

Figure 2. continued

Unsorted sample bibliography 50 columns wide.
Keywords included.

[1] Clay,D.,
Clay,H. (1979)
Bibliography of the hake family (Pisces: Merlucciidae). ICNAF Res. Doc. 79/VI/78. Serial No. 5381.

[2] Stearns,S.C. (1976)
Life-history tactics: A review of the ideas. Quar. Rev. Biol. 51(1):3-47.
(evolution reproduction mortality)

[3] Pieterse,A.J.H.,
Roos,J.C.,
Roos,K.I.,
Piennaar,C. (1986)
Preliminary observations on cross-channel and vertical heterogeneity in environmental and algological parameters in the \ al River at Balkfontein, South Africa. Water SA 12(4):173-184.
(temperature TDS turbidity)

[4] Avers,G.S.,
Hoopingarner,R.A. (1986)
The potential for fixed-land honey production. Amer. Bee J. 126(12):805-808.
(forage bees nectar)

[5] Avers,G.S.,
Hoopingarner,R.A. (1987)
Research imperatives for fixed-land honey production. Amer. Bee J. 127(1):39-41.
(forage herbicides)

[6] Clay,D.,
Kenchington,T.J. (1986)
World bibliography of the redfishes and rockfishes (Sebastinae, Scorpaenidae). Can. Tech. Rep. Fish. Aquat. Sci. 1429: iii +303pp.

[7] Dabrowski,K.,
Bardega,R. (1984)
Mouth size and predicted food size preferences of three cyprid fish species. Aquaculture 40:41-46.

[8] Dabrowski,K. (1984)
The feeding of fish larvae: present "state of the art" and perspectives. Reprod. Nutn. Develop. 24(6):807-833.
(growth mortality)

[9] Kettleborough,I. (1982)
Nevada fortran: Programmer s reference manual. Ellis Computing Inc., San Francisco, USA. 166pp.
(compiler computer language 8-bit Z-80 CPM)

[10] Kettleborough,I.D. (1985)
Utah fortran: Programmer s reference manual. Ellis Computing Inc., Reno, Nevada, USA. 222pp.
(MS-DOS PC-DOS compiler language 16-bit computer)

[11] ZZZZZZZZZZ (9999)
END OF FILE

[12] Wells,P.G. (1977)
A selected bibliography on rainbow trout, Salmo gairdneri Richardson, with particular reference to studies with aquatic toxicants. Can.Env.Prot. Service, Econ.Tech.Rev.Rept. EPS-3-AR-77-1. 109pp.

[13] Marshall,K.E.,
Kelsher,J.J. (1970)
A bibliography of the lake trout, Cristivomer namaycush (Walbaum): 1929-1960. F.R.B.Can.Tech. Rept. 176: iv + 60pp.

[14] Van der Waal,B.C.W. (1976)
[An ecological study of the fish of the east Caprivi, with sampling from the bantu fishery.] Ph.D. Thesis, Rand Africaans Univer. Johannesburg, Republic of South Africa. (In Africaans).
(age swamp catfish Clarias)

[15] El-Bolock,A.R. (1972)
Use of vertebrae for determination of age and growth of Nile catfish, Clarias lazera. U.A.R. Bull. Inst. Ocean. Fish. 2:53-82.

[16] ZZZZZZZZZZ (9999)
END OF FILE.

Figure 3. "KWIC" index of sample bibliography in both sorted and unsorted forms. Keywords included.

Unsorted: first 6 references only

```
1 ..... $Clay,D., $Clay,H. $Bibliography of the h
1 ..... $Clay,D., $Clay,H. $Bibliography of the hake family
1 ..... $Clay,D., $Clay,H. Bibliography of the hake family (Pisces :
1 ..... $Clay,D., $Clay,H. Bibliography of the hake family (Pisces : Merluccidae).
1 ..... $Clay,D., $Clay,H. Bibliography of the hake family (Pisces : Merluccidae).
1 ..... $Clay,D., $Clay,H. Bibliography of the hake family (Pisces : Merluccidae).
1 ..... $Clay,D., $Clay,H. Bibliography of the hake family (Pisces : Merluccidae).
2 ..... $Stearns,S.C. $Life - history tactics : A
2 ..... $Stearns,S.C. Life - history tactics : A review of the
2 ..... $Stearns,S.C. Life - history tactics : A review of the ideas.
2 ..... $Stearns,S.C. Life - history tactics : A review of the ideas. (evoluti
2 ..... $Stearns,S.C. Life - history tactics : A review of the ideas. (evolution) (reprodu
2 ..... $Stearns,S.C. Life - history tactics : A review of the ideas. (evolution) (reproduction) (mortal
2 ..... $Stearns,S.C. Life - history tactics : A review of the ideas. (evolution) (reproduction) (mortality)
2 ns,S.C. Life - history tactics : A review of the ideas. (evolution) (reproduction) (mortality)
2  history tactics : A review of the ideas. (evolution) (reproduction) (mortality)
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I.,
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. $Pre
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. $Preliminary obs
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. $Preliminary observations on
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. Preliminary observations on cross - chann
3 . $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. Preliminary observations on cross - channel and verti
3 H., $Roos,J.C., $Roos,K.I., $Piennaar,C. Preliminary observations on cross - channel and vertical heterogeneit
3 s,J.C., $Roos,K.I., $Piennaar,C. Preliminary observations on cross - channel and vertical heterogeneity in env
3 s,K.I., $Piennaar,C. Preliminary observations on cross - channel and vertical heterogeneity in environmental a
3 Piennaar,C. Preliminary observations on cross - channel and vertical heterogeneity in environmental and algolo
3 iminary observations on cross - channel and vertical heterogeneity in environmental and algological parameters
3 ns on cross - channel and vertical heterogeneity in environmental and algological parameters in the Vaal River
3 - channel and vertical heterogeneity in environmental and algological parameters in the Vaal River at Balkfonte
3 ical heterogeneity in environmental and algological parameters in the Vaal River at Balkfontein , South Africa.
3 heterogeneity in environmental and algological parameters in the Vaal River at Balkfontein , South Africa. (tem
3 eity in environmental and algological parameters in the Vaal River at Balkfontein , South Africa. (temperature)
3 nmental and algological parameters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbid
3 l and algological parameters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbidity)
3 ological parameters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbidity)
3 eters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbidity)
3 in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbidity)
4 ..... $Avers,G.S., $Hoopingarner,R.A. $The pote
4 ..... $Avers,G.S., $Hoopingarner,R.A. $The potential for fix
4 ..... $Avers,G.S., $Hoopingarner,R.A. The potential for fixed - land honey producti
4 ..... $Avers,G.S., $Hoopingarner,R.A. The potential for fixed - land honey production. (forage) (
4 ..... $Avers,G.S., $Hoopingarner,R.A. The potential for fixed - land honey production. (forage) (bees) (n
4 ..... $Avers,G.S., $Hoopingarner,R.A. The potential for fixed - land honey production. (forage) (bees) (nectar
4 $Avers,G.S., $Hoopingarner,R.A. The potential for fixed - land honey production. (forage) (bees) (nectar)
4 $Hoopingarner,R.A. The potential for fixed - land honey production. (forage) (bees) (nectar)
4 rner,R.A. The potential for fixed - land honey production. (forage) (bees) (nectar)
4 A. The potential for fixed - land honey production. (forage) (bees) (nectar)
5 ..... $Avers,G.S., $Hoopingarner,R.A. $Research
5 ..... $Avers,G.S., $Hoopingarner,R.A. $Research imperatives
5 ..... $Avers,G.S., $Hoopingarner,R.A. Research imperatives for fixed - land hon
5 ..... $Avers,G.S., $Hoopingarner,R.A. Research imperatives for fixed - land honey produc
5 ..... $Avers,G.S., $Hoopingarner,R.A. Research imperatives for fixed - land honey production. (forage) (
5 ... $Avers,G.S., $Hoopingarner,R.A. Research imperatives for fixed - land honey production. (forage) (herbicid
5 Avers,G.S., $Hoopingarner,R.A. Research imperatives for fixed - land honey production. (forage) (herbicides)
5 G.S., $Hoopingarner,R.A. Research imperatives for fixed - land honey production. (forage) (herbicides)
5 garner,R.A. Research imperatives for fixed - land honey production. (forage) (herbicides)
5 A. Research imperatives for fixed - land honey production. (forage) (herbicides)
6 ..... $Clay,D., $Kenchington,T.J. $World biblio
6 ..... $Clay,D., $Kenchington,T.J. $World bibliography of
6 ..... $Clay,D., $Kenchington,T.J. World bibliography of the redfishes and r
6 ..... $Clay,D., $Kenchington,T.J. World bibliography of the redfishes and rockfis
6 ..... $Clay,D., $Kenchington,T.J. World bibliography of the redfishes and rockfishes (Sebastinae , Sc
6 ^$Clay,D., $Kenchington,T.J. World bibliography of the redfishes and rockfishes (Sebastinae , Scorpaenidae).
6 Kenchington,T.J. World bibliography of the redfishes and rockfishes (Sebastinae , Scorpaenidae).
6 .J. World bibliography of the redfishes and rockfishes (Sebastinae , Scorpaenidae).
```

FIGURE 3. continued

Sorted: all 15 references

5 ..... $Myers,G.S., $Hoopingarner,R.A. $Research
4 ..... $Myers,G.S., $Hoopingarner,R.A. $The pote
7 ..... $Dabrowski,K., $Bardega,R. $Mouth size and predicted foo
1 ..... $Clay,D., $Clay,H. $Bibliography of the h
6 ..... $Clay,D., $Henchington,T.J. $World biblio
1 ..... $Clay,D., $Clay,H. $Bibliography of the hake family
8 ..... $Dabrowski,K. $The feeding of fish larvae
7 ..... $Dabrowski,K., $Bardega,R. $Mouth size an
12 ..... $El-Bolock,A.R. $Use of vertebrae for det
5 ..... $Myers,G.S., $Hoopingarner,R.A. $Research imperatives
4 ..... $Myers,G.S., $Hoopingarner,R.A. $The potential for fix
13 ..... $Marshall,K.E., $Weleher,J.J. $A bibliography of the lake
9 ..... $Clay,D., $Henchington,T.J. $World bibliography of
9 ..... $Wattleborough,I. $Nevada fortran : Progr
10 ..... $Wattleborough,I.D. $Utah fortran : Progr
13 ..... $Marshall,K.E., $Weleher,J.J. $A bibliogr
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. $Preliminary observations on
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I..
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. $Pre
3 ..... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C. $Preliminary obs
2 ..... $Stearns,S.C. $Life - history tactics : A
14 ..... $Van der Waal,B.C.W. $[An ecological stud
12 ..... $Wells,P.S. $A selected bibliography on r
10 ogrammer s reference manual. (MS-DOS) (PC-DOS) (compiler) (language) (16-bit) (computer)
9 an : Programmer s reference manual. (compiler) (computer) (language) (8-bit) (Z-80) (CPM)
3 l and algological parameters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbidity)
15 ..... $El-Bolock,A.R. Use of vertebrae for determination of age and growth of Nile catfish , Clarias
14 the fish of the east Caprivi , with sampling from the bantu fishery. (age) (swamp) (catfish) (Clarias)
3 ns on cross - channel and vertical heterogeneity in environmental and algological parameters in the Vaal River
12 alen gairdneri Richardson , with particular reference to studies with aquatic toxicants.
3 eity in environmental and algological parameters in the Vaal River at Balkfontein , South Africa. (temperature)
14 ogical study of the fish of the east Caprivi , with sampling from the bantu fishery. (age) (swamp) (catfish) (C
4 rner,R.A. The potential for fixed - land honey production. (forage) (bees) (nectar)
1 ..... $Clay,D., $Clay,H. Bibliography of the hake family (Pisces :
13 ..... $Marshall,K.E., $Weleher,J.J. A bibliography of the lake trout , Cristivo
5 ..... $Clay,D., $Henchington,T.J. World bibliography of the redfishes and rockfis
12 ..... $Wells,P.S. A selected bibliography on rainbow trout , Salmo gai
14 .. $Van der Waal,B.C.W. [An ecological study of the fish of the east Caprivi , with sampling from the bantu fi
15 ck,A.R. Use of vertebrae for determination of age and growth of Nile catfish , Clarias lazera.
14 e east Caprivi , with sampling from the bantu fishery. (age) (swamp) (catfish) (Clarias)
3 s,J.C., $Roos,K.I., $Piennaar,C. Preliminary observations on cross - channel and vertical heterogeneity in env
15 se of vertebrae for determination of age and growth of Nile catfish , Clarias lazera.
14 rivi , with sampling from the bantu fishery. (age) (swamp) (catfish) (Clarias)
9 , $Wattleborough,I. Nevada fortran : Programmer s reference manual. (compiler) (computer) (language) (8-bit) (
10 .D. Utah fortran : Programmer's reference manual. (MS-DOS) (PC-DOS) (compiler) (language) (16-bit) (computer)
10 s reference manual. (MS-DOS) (PC-DOS) (compiler) (language) (16-bit) (computer)
9 rough,I. Nevada fortran : Programmer's reference manual. (compiler) (computer) (language) (8-bit) (Z-80) (CPM)
9 's reference manual. (compiler) (computer) (language) (8-bit) (Z-80) (CPM)
13 ... $Marshall,K.E., $Weleher,J.J. A bibliography of the lake trout , Cristivomer namaycush (Walbaum) : 1920 -
3 H., $Roos,J.C., $Roos,K.I., $Piennaar,C. Preliminary observations on cross - channel and vertical heterogeneit
7 $Bardega,R. Mouth size and predicted food size preferences of three cyprid fish species.
15 ..... $El-Bolock,A.R. Use of vertebrae for determination of age and growth of Nile c
14 ..... $Van der Waal,B.C.W. [An ecological study of the fish of the east Caprivi , with sampling from the ban
14 ..... $Van der Waal,B.C.W. [An ecological study of the fish of the east
3 iminary observations on cross - channel and vertical heterogeneity in environmental and algological parameters
2 ..... $Stearns,S.C. Life - history tactics : A review of the ideas. (evolution) (reproduction) (mortality)
1 ..... $Clay,D., $Clay,H. Bibliography of the hake family (Pisces : Merluccidae).
8 ..... $Dabrowski,K. The feeding of fish larvae : present "state o
8 ..... $Dabrowski,K. The feeding of fish larvae : present "state of the art"
14 ..... $Van der Waal,B.C.W. [An ecological study of the fish of the east Caprivi , with sampling
7 ga,R. Mouth size and predicted food size preferences of three cyprid fish species.
14 study of the fish of the east Caprivi , with sampling from the bantu fishery. (age) (swamp) (catfish) (Clarias
4 ..... $Myers,G.S., $Hoopingarner,R.A. The potential for fixed - land honey production. (forage) (
5 ..... $Myers,G.S., $Hoopingarner,R.A. Research imperatives for fixed - land honey production. (forage) (
7 ..... $Dabrowski,K., $Bardega,R. Mouth size and predicted food size preferences of three cyprid fis
4 $Hoopingarner,R.A. The potential for fixed - land honey production. (forage) (bees) (nectar)
9 ..... $Wattleborough,I. Nevada fortran : Programmer s reference manual.
10 ..... $Wattleborough,I.D. Utah fortran : Programmer s reference manual.

FIGURE 3. continued

```
12 ..... $Wells,P.G.  A selected bibliography on rainbow trout , Salmo gairdneri Richardson , with particular re
15 ..... $El-Bolock,A.R.  Use of vertebrae for determination of age and growth of Nile catfish , Clarias lazera.
 8 eeding of fish larvae : present "state of the art" and perspectives. (growth) (mortality)
 1 ............................. $Clay,D., $Clay,H.  Bibliography of the hake family (Pisces : Merluccidae).
 5 A.  Research imperatives for fixed - land honey production. (forage) (herbicides)
 3 Piennaar,C.  Preliminary observations on cross - channel and vertical heterogeneity in environmental and algolo
 2 ..................................................... $Stearns,S.C.  Life - history tactics : A review of the ideas.
 4 ..... $Ayers,G.S., $Hoopingarner,R.A.  The potential for fixed - land honey production. (forage) (bees) (nectar
 5 Ayers,G.S., $Hoopingarner,R.A.  Research imperatives for fixed - land honey production. (forage) (herbicides)
 2 ............. $Stearns,S.C.  Life - history tactics : A review of the ideas. (evolution) (reproduction) (mortal
 5 ............................. $Ayers,G.S., $Hoopingarner,R.A.  Research imperatives for fixed - land honey produc
13 ................ $Marshall,K.E., $Kelsher,J.J.  A bibliography of the lake trout , Cristivomer namaycush (Walba
 4 .......... $Ayers,G.S., $Hoopingarner,R.A.  The potential for fixed - land honey production. (forage) (bees) (n
 5 ... $Ayers,G.S., $Hoopingarner,R.A.  Research imperatives for fixed - land honey production. (forage) (herbicid
10 ortran : Programmer's reference manual. (MS-DOS) (PC-DOS) (compiler) (language) (16-bit) (computer)
 9 evada fortran : Programmer's reference manual. (compiler) (computer) (language) (8-bit) (Z-80) (CPM)
 8 ..................................... $Dabrowski,K.  The feeding of fish larvae : present "state of the art" and p
15 rtebrae for determination of age and growth of Nile catfish , Clarias lazera.
 2 ..................................................... $Stearns,S.C.  Life - history tactics : A review of the
 9 ........... $Kettleborough,I.  Nevada fortran : Programmer's reference manual. (compiler) (computer) (language)
10 ........... $Kettleborough,I.D.  Utah fortran : Programmer's reference manual. (MS-DOS) (PC-DOS) (compiler) (lan
 1 ....... $Clay,D., $Clay,H.  Bibliography of the hake family (Pisces : Merluccidae).
 2 history tactics : A review of the ideas. (evolution) (reproduction) (mortality)
 8 fish larvae : present "state of the art" and perspectives. (growth) (mortality)
 7 ........................................... $Dabrowski,K., $Bardega,R.  Mouth size and predicted food size prefer
10 . $Kettleborough,I.D.  Utah fortran : Programmer's reference manual. (MS-DOS) (PC-DOS) (compiler) (language) (1
13 l,K.E., $Kelsher,J.J.  A bibliography of the lake trout , Cristivomer namaycush (Walbaum) : 1929 - 1960.
 4 A.  The potential for fixed - land honey production. (forage) (bees) (nectar)
15 -Bolock,A.R.  Use of vertebrae for determination of age and growth of Nile catfish , Clarias lazera.
 3 . $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C.  Preliminary observations on cross - channel and verti
 3 - channel and vertical heterogeneity in environmental and algological parameters in the Vaal River at Balkfonte
10 borough,I.D.  Utah fortran : Programmer's reference manual. (MS-DOS) (PC-DOS) (compiler) (language) (16-bit) (c
 8 owski,K.  The feeding of fish larvae : present "state of the art" and perspectives. (growth) (mortality)
 1 ............... $Clay,D., $Clay,H.  Bibliography of the hake family (Pisces : Merluccidae).
 4 ............................. $Ayers,G.S., $Hoopingarner,R.A.  The potential for fixed - land honey producti
 7 ........................... $Dabrowski,K., $Bardega,R.  Mouth size and predicted food size preferences of three
 7 ....... $Dabrowski,K., $Bardega,R.  Mouth size and predicted food size preferences of three cyprid fish species.
 3 ............... $Pieterse,A.J.H., $Roos,J.C., $Roos,K.I., $Piennaar,C.  Preliminary observations on cross - chann
 4 $Ayers,G.S., $Hoopingarner,R.A.  The potential for fixed - land honey production. (forage) (bees) (nectar)
 5 G.S., $Hoopingarner,R.A.  Research imperatives for fixed - land honey production. (forage) (herbicides)
 5 garner,R.A.  Research imperatives for fixed - land honey production. (forage) (herbicides)
 9 ................................... $Kettleborough,I.  Nevada fortran : Programmer's reference manual. (compiler)
10 ................................... $Kettleborough,I.D.  Utah fortran : Programmer's reference manual. (MS-DOS) (
12 ............................. $Wells,P.G.  A selected bibliography on rainbow trout , Salmo gairdneri Richardso
 5 ................ $Clay,D., $Kenchington,T.J.  World bibliography of the redfishes and rockfishes (Sebastinae , Sc
 9 ..................... $Kettleborough,I.  Nevada fortran : Programmer's reference manual. (compiler) (computer) (
10 ..................... $Kettleborough,I.D.  Utah fortran : Programmer's reference manual. (MS-DOS) (PC-DOS) (compi
12 raphy on rainbow trout , Salmo gairdneri Richardson , with particular reference to studies with aquatic toxican
 2 ns,S.C.  Life - history tactics : A review of the ideas. (evolution) (reproduction) (mortality)
 5 ..................................... $Ayers,G.S., $Hoopingarner,R.A.  Research imperatives for fixed - land hon
 2 ............................ $Stearns,S.C.  Life - history tactics : A review of the ideas. (evolution) (reprodu
12 ells,P.G.  A selected bibliography on rainbow trout , Salmo gairdneri Richardson , with particular reference to
 3 heterogeneity in environmental and algological parameters in the Vaal River at Balkfontein , South Africa. (tem
 6 $Clay,D., $Kenchington,T.J.  World bibliography of the redfishes and rockfishes (Sebastinae , Scorpaenidae).
12 .............. $Wells,P.G.  A selected bibliography on rainbow trout , Salmo gairdneri Richardson , with particu
14 l,B.C.W.  (An ecological study of the fish of the east Caprivi , with sampling from the bantu fishery. (age) (s
 6 .J.  World bibliography of the redfishes and rockfishes (Sebastinae , Scorpaenidae).
 6 Kenchington,T.J.  World bibliography of the redfishes and rockfishes (Sebastinae , Scorpaenidae).
 7 ..................................... $Dabrowski,K., $Bardega,R.  Mouth size and predicted food size preferences
 7 ............ $Dabrowski,K., $Bardega,R.  Mouth size and predicted food size preferences of three cyprid fish spe
 3 nental and algological parameters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbid
 7  Mouth size and predicted food size preferences of three cyprid fish species.
14 sh of the east Caprivi , with sampling from the bantu fishery. (age) (mesmo) (catfish) (Clarias)
 2 ..................................... $Stearns,S.C.  Life - history tactics : A review of the ideas. (evoluti
 3 eters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbidity)
 3 ological parameters in the Vaal River at Balkfontein , South Africa. (temperature) (TDS) (turbidity)
12 rgneri Richardson , with particular reference to studies with aquatic toxicants.
13 ............ $Marshall,K.E., $Kelsher,J.J.  A bibliography of the lake trout , Cristivomer namaycush (Walbaum) :
12 ..................... $Wells,P.G.  A selected bibliography on rainbow trout , Salmo gairdneri Richardson , with

                                .............purposely cutoff at the V's.................
```

APPENDIX I

Source listing for the program (BIBIN.FOR).

```
***** UTAH Fortran 1.0 (Mod 2) ** Compiling File: D:BIBIN.FOR *****

0001 $OPTIONS L=150,S=150,E,X
     C
     C*****************************************************************************
     C*    BIBLIOGRAPHY INPUT PROGRAM                                             *
     C*                                                                           *
     C*    THIS PROGRAM ALLOWS A BIBLIOGRAPHY FILE TO BE INPUT IN A STANADRD      *
     C*    FORMAT FOR MERGING / SORTING AND PRINTING. THE PROGRAM IS IN FORTRAN   *
     C*    IV (NEVADA FORTRAN) AND INTERFACES WITH TWO OTHER MODULES IN THE       *
     C*    BIB???.??? PACKAGE. THE SORT IS CONDUCTED BY MICRO-PRO s SUPERSORT.    *
     C*                                                                           *
     C*    INPUT CAN BE TO AN EXISTING FILE OR TO A NEWLY CREATED FILE. AUTHOR    *
     C*    INPUT IS EDITED TO A STANDARD FORMAT.                                  *
     C*                                                                           *
     C*                PROGRAM BY:    DOUGLAS CLAY                              *
     C*                               MARINE FISHERIES DIVISION                 *
     C*                               GULF FISHERIES CENTER                     *
     C*                               DEPARTMENT OF FISHERIES AND OCEANS        *
     C*                               P.O.BOX 5030, MONCTON, N.B., CANADA.      *
     C*                                                                         *
     C*                (version 1 : June 1978 : CDC Fortran IV)                 *
     C*    ORIGINAL PROGRAM CALLED (UPBIB)                                      *
     C*                VERSION :      2.0       DATE :  22/12/84               *
     C*    RE-WRITTEN FOR 8-BIT MICRO-COMPUTERS                                 *
     C*                        :      2.1       DATE :  10/03/86               *
     C*    PROGRAM CONVERTED TO 16 bit MSDOS COMPUTER USING UTAH FORTRAN.       *
     C*                        :      3.0       DATE :  19/01/87               *
     C*                                                                         *
     C*    PROGRAM NOW HANDLES ADDITIONAL KEYWORDS ADDED AT THE END OF THE      *
     C*    REFERENCE. THESE ARE STORED IN GROUPS OF UP TO 40 CHARACTERS EACH     *
     C*    AND BOUNDED BY CURLY BRACKETS ( ... ).                               *
     C*****************************************************************************
     C
0002       DIMENSION FNAME(3), AUTHOR(4), TITLE(10), HOLD(4)
0003       REAL      KEYWRD(49)
0004       DATA BLANK /'       '/, QUIT /'END '/, FINISH /' end  '/, Y /'Y '/,
          >     YY/'y'/, SPACE /' '/, CURLR /'} '/
     C
0005       IOR = 0
0006       IOW = 1
0007       IOUT = 2
     C
0008       WRITE (IOW,1247)
0009 1247  FORMAT (//////,'     ----   BIBLIOGRAPHIC INPUT   ---- ',/////,
          1'LAST REVISION  22/01/87  version 3.0'/////,10X,'...c...'/////)
     C
0010 1227  WRITE (IOW,1246)
0011 1246  FORMAT (//' NAME OUTPUT FILE OR <CR> TO DEFAULT OF ',
          >'<BIBLIO.DAT> ',Z)
0012       READ (IOR,1234) FNAME
0013 1234  FORMAT (10A6)
     C
0014       IER = 0
0015         IF (FNAME(1).EQ.BLANK) THEN
0016            CALL OPEN (2,'BIBLIO.DAT',IER)
0017            IF (IER.NE.0)  CALL OPEN (2,'BIBLIO.DAT')
0018          ELSE
0019            CALL OPEN (2,FNAME,IER)
0020            IF (IER .NE. 0) GOTO 1224
0021            WRITE (IOW,1235)
0022 1235       FORMAT (' FILE ALREADY EXISTS - OK TO READ TO END (Y)es
          >               (N)o ' , Z)
0023            READ (IOR,1237) K
0024 1237       FORMAT (A1)
0025            IF (K .EQ. Y .OR. K .EQ. YY) GOTO 1225
0026            CALL CLOSE (2)
0027            GOTO 1227
```

```
0028 1224        CALL OPEN (2,FNAME)
0029 1225        CONTINUE
0030           ENDIF
     C
0031         REWIND IOUT
0032         EMPTY=0
     C
0033         IF (IER.EQ.0) THEN
0034 1           READ (IOUT,6,END=11) YEAR , AUTHOR
0035             IF (EMPTY.EQ.0) THEN
0036                 EMPTY=1
0037                 WRITE (IOW,*) ' READING TO END OF BIBLIOGRAPHY FILE
0038             ENDIF
0039             IF (YEAR .NE. '9999') GOTO 1
0040             GOTO 13
0041         ELSE
0042             WRITE (IOW,*) ' NEW FILE'    .
0043         ENDIF
     C
0044 11      IF (IER.EQ.0) WRITE (IOUT,12)
0045 12                    FORMAT ('\0A\')
0046 13      IREF = 0
0047 2       WRITE (IOW,*) ' YEAR    or end .....
0048         READ (IOR,3) YEAR
0049 3       FORMAT (A4)
     C
0050            IF (YEAR.NE.QUIT .AND. YEAR.NE.FINISH) THEN
0051            IREF = IREF + 1
     C
0052               FIRST = 0
0053 4             WRITE (IOW,*) ' AUTHOR    or <CR> .....
0054               WRITE (IOW,41)
0055 41            FORMAT ('<',22(' '),'>')
0056               READ (IOR,1234) HOLD , TEST
     C
0057               IF (TEST.NE.BLANK) THEN
0058                  WRITE (IOW,*) ' WARNING: AUTHORS CAN ONLY BE ENTERED 24
0059                  WRITE (IOW,*) '           CHARACTERS AT A TIME; RE-ENTER
0060                  GOTO 4
0061               ENDIF
     C
0062               IF (HOLD(1).NE.BLANK) THEN
     C
0063                  IF (FIRST.EQ.0) THEN
0064                      FIRST = FIRST + 1
0065                      DO 5 I = 1 , 4
0066 5                    AUTHOR(I) = HOLD(I)
0067                      GOTO 4
0068                  ENDIF
     C
0069                  CALL EDIT (AUTHOR,1)
     C
0070                  WRITE (IOUT,6) YEAR , AUTHOR
0071 6                FORMAT (A4,2X,4A6)
     C
0072                     IF (FIRST.EQ.1) YEAR = BLANK
     C
0073                  DO 61 I = 1 , 4
0074 61               AUTHOR(I) = HOLD(I)
0075                  GOTO 4
0076               ENDIF
     C
0077               FIRST = 0
0078 7             WRITE (IOW,*) ' TITLE    or <CR> .....
0079               WRITE (IOW,71)
0080 71            FORMAT ('<',48(' '),'>')
0081               READ (IOR,8) TITLE , TEST
0082 8             FORMAT (10A5,A6)
     C
```

```
0083              IF (TEST.NE.BLANK) THEN
0084                 WRITE (IOW,*) ' WARNING: TITLE CAN ONLY BE ENTERED 50
0085                 WRITE (IOW,*) '          CHARACTERS AT A TIME: RE-ENTER
0086                 GOTO 7
0087              ENDIF
     C
0088              IF (TITLE(1).NE.BLANK) THEN
     C
0089                  IF (FIRST.EQ.0) CALL EDIT (AUTHOR,-1)
     C
0090                  WRITE (IOUT,9) YEAR , AUTHOR , TITLE
0091 9                FORMAT (A4,2X,4A6,10A5)
     C
0092                  IF (FIRST.EQ.0) THEN
0093                      FIRST = FIRST + 1
0094                      DO 91 I = 1 , 4
0095 91                   AUTHOR (I) = BLANK
0096                      CALL EDIT (AUTHOR,1)
0097                      YEAR = BLANK
0098                      GOTO 7
0099                  ELSE
0100                      GOTO 7
0101                  ENDIF
0102              ENDIF
     C
0103 10           WRITE (IOW,*) 'KEY WORDS     or (CR) ......
0104              WRITE (IOW,102)
0105 102          FORMAT ('(',46(' '),'$')
0106              READ (IOR,104) (KEYWRD(I),I = 1,48) , TEST
0107 104          FORMAT (48A1,A5)
     C
0108              IF (TEST.NE.BLANK) THEN
0109                 WRITE (IOW,*) ' WARNING: KEYWORDS CAN ONLY BE ENTERED 48
0110                 WRITE (IOW,*) '          CHARACTERS AT A TIME: RE-ENTER
0111                 GOTO 10
0112              ENDIF
     C
0113 105          IF (KEYWRD(1).NE.SPACE) THEN
     C
0114                 DO 106 I = 48 , 1 , -1
0115                 J = I + 1
0116 106             IF (KEYWRD(I).NE.SPACE) GOTO 108
0117 108             KEYWRD(J) = CURLR
0118                 WRITE (IOUT,109) YEAR , AUTHOR , (KEYWRD(I),I=1,J)
0119 109             FORMAT (A4,2X,4A6, (',49A1')
0120                 GOTO 10
0121              ENDIF
     C
0122              IF (KEYWRD(2) .NE. SPACE) THEN
0123                 DO 110 I = 1,47
0124 110             KEYWRD(I) = KEYWRD(I+1)
0125                 GOTO 105
0126              ENDIF
     C
0127              IF (FIRST.EQ.0) IREF = IREF - 1
0128              GOTO 2
0129           ENDIF
     C
0130       WRITE (IOUT,*) '9999  ZZZZZZZZZZ                END OF FILE.
0131       CALL CLOSE (IOUT)
0132       WRITE (IOW,*) ' TOTAL NUMBER OF REFERENCES INPUT: ',IREF
0133       STOP
0134       END
** Generated Code =  2899 (Decimal), 0B53 (Hex) Bytes
```

```
0001 $OPTIONS X,E,L=100,S=100
0002       SUBROUTINE EDIT (HOLD,TYPE)
     C
     C*****************************************************************************
     C*                                                                           *
     C*    SUBROUTINE TO CHECK (and standardize) THE AUTHOR LISTING STYLE         *
     C*    IN COLUMNS 7 TO 30 TO ENSURE UNIFORM FORMATTING.                       *
     C*                                                                           *
     C*                PROGRAM BY:    DOUGLAS CLAY                              *
     C*                               MARINE FISHERIES DIVISION                 *
     C*                               GULF FISHERIES CENTER                     *
     C*                               DEPARTMENT OF FISHERIES AND OCEANS        *
     C*                               P.O.BOX 5030, MONCTON, N.B., CANADA.      *
     C*                                                                         *
     C*                (version 1 : Aug 1980 : CDC Fortran IV)                  *
     C*     ORIGINALLY A SEPERATE PROGRAM CALLED (EDBIB)                        *
     C*                VERSION :      2.0       DATE :  24/12/84               *
     C*     RE-WRITTEN FOR 8-BIT MICRO-COMPUTERS                                *
     C*                        :      2.1       DATE :  10/02/86               *
     C*    PROGRAM CONVERTED TO 16 bit MSDOS COMPUTER USING UTAH FORTRAN.       *
     C*****************************************************************************
     C
0003       DIMENSION HOLD (4) , STRIP (24)
0004       DATA BLANK /' '/ , COMMA /','/ , PERIOD /'.'/
     C
0005       DECODE (HOLD,24,1) STRIP
0006 1     FORMAT (24A1)
     C
0007       DO 2 I = 24 , 1 , -1
0008       IF (STRIP(I).NE.BLANK) THEN
0009           LAST = I
0010           LESS1 = I - 1
0011           GOTO 3
0012       ENDIF
0013 2     CONTINUE
     C
0014 3     BEGIN = 0
0015       DO 5 I = 1 , LAST
     C       FINDING FIRST OCCURRENCE OF A PUNCTUATION MARK ( , or . )
0016            IF (BEGIN.EQ.0) THEN
0017              IF (STRIP(I).EQ.COMMA .OR. STRIP(I).EQ.PERIOD) THEN
0018                    I = I - 2
0019                    BEGIN = 1
0020                    GOTO 5
0021              ELSE
0022                   GOTO 5
0023              ENDIF
0024            ENDIF
     C       REMOVING ALL BLANKS AFTER FAMILY NAME
0025          IF (STRIP(I).EQ.BLANK) THEN
0026              DO 4 J = I , LESS1
0027 4            STRIP (J) = STRIP (J+1)
0028              STRIP (LAST) = BLANK
0029          ENDIF
0030 5     CONTINUE
     C
     C     WHEN NOT THE LAST OR ONLY AUTHOR - THEN ADD COMMA IF OMITTED (TYPE   0)
0031       IF (TYPE.GT.0) THEN
0032           DO 6 I = 24 , 1 , -1
0033           IF (STRIP(I).NE.BLANK) THEN
0034              IF (STRIP(I).EQ.PERIOD) STRIP(I+1) = COMMA

0035              GOTO 9
0036           ENDIF
0037 6         CONTINUE
     C     WHEN THE LAST OR ONLY AUTHOR - THEN REMOVE COMMA IF ADDED (TYPE   0)
0038       ELSE
```

```
0039            DO 7 I = 24 , 1 , -1
0040            IF (STRIP(I).NE.BLANK) THEN
0041               IF (STRIP(I).EQ.COMMA) STRIP(I) = BLANK
0042               GOTO 8
0043            ENDIF
0044 7          CONTINUE
0045         ENDIF
     C
0046 8       CONTINUE
     C
0047         ENCODE (HOLD,24,9) (STRIP(I),I=1,24)
0048 9       FORMAT (4(6A1))
     C
0049         RETURN
0050         END
++ Generated Code =   840 (Decimal), 0348 (Hex) Bytes

       No Compile errors
```

APPENDIX II

Source listing for the program BIBPRT.FOR>.

```
***** UTAH Fortran 1.0 (Mod 2) ** Compiling File: D:BIBPRT.FOR *****

0001 $OPTIONS L=150,S=150,E,X
     C
     C*************************************************************************
     C*    BIBLIOGRAPHY PRINTING PROGRAM                                      *
     C*                                                                       *
     C*    THIS PROGRAM ALLOWS A BIBLIOGRAPHY FILE TO BE PRINTED FROM THE     *
     C*    STANDARD FORMAT USED FOR MERGING/SORTING (AFTER IT HAS BEEN SORTED)*
     C*    OR FROM THE STANDARD INPUT DATA FORMAT.                            *
     C*                                                                       *
     C*    OUTPUT CAN BE ONE OF THREE FORMS: TWO ARE FOR BIBLIOGRAPHICAL      *
     C*    OUTPUT WHILE THE THIRD IS IN THE INPUT FORMAT (ie NORMALLY USED    *
     C*    TO RETURN A FILE, AFTER SORTING, TO ITS ORIGINAL FORMAT).          *
     C*                                                                       *
     C*    THIS PROGRAM IS WRITTEN IN FORTRAN IV OR FORTRAN 66 (CP/M - Z80    *
     C*    NEVADA FORTRAN) AND INTERFACES WITH TWO OTHER MODULES IN THE        *
     C*    BIB???.??? PACKAGE. THE SORT IS CONDUCTED BY MICRO-PRO'S SUPERSORT.*
     C*                                                                       *
     C*                PROGRAM BY:     DOUGLAS CLAY                           *
     C*                                MARINE FISHERIES DIVISION              *
     C*                                GULF FISHERIES CENTER                  *
     C*                                DEPARTMENT OF FISHERIES AND OCEANS     *
     C*                                P.O.BOX 5030, MONCTON, N.B., CANADA.   *
     C*                                                                       *
     C*                (version 1 : April    1975 : ICL Fortran IV)           *
     C*                (version 2 : Novemeber 1977 : CDC Fortran IV)          *
     C*    ORIGINAL PROGRAM CALLED (PROBIB)                                   *
     C*                VERSION :       3.0       DATE :  30/01/85           *
     C*    RE-WRITTEN FOR 8-BIT MICRO-COMPUTERS                               *
     C*                          :     3.1       DATE :  10/03/85           *
     C*    PROGRAM CONVERTED TO 16 bit MSDOS COMPUTER USING UTAH FORTRAN.     *
     C*                          :     4.0       DATE :  20/01/87           *
     C*                                                                       *
     C*    PROGRAM NOW HANDLES ADDITIONAL KEYWORDS ADDED AT THE END OF THE    *
     C*    REFERENCE. THESE ARE STORED IN GROUPS OF UP TO 48 CHARACTERS EACH   *
     C*    AND BOUNDED BY CURLY BRACKETS { ... }.                             *
     C*************************************************************************
     C
0002      DIMENSION  FNAME(3), HOLD(4), TITLE(50), STRING(496), AUTHOR(24),
     )           YEAR(4)
0003      REAL KEY , EXIST
0004      COMMON /BIB/ STRING, IOW, IOR, IREF  SORT, WIDE, YEAR, KEY
0005      COMMON /GET/ HOLD , TITLE , ORDER , LINENO , AUTHOR , TOTAUT
0006      DATA BLANK /'    '/ , Y / 'Y' / , YY /'Y' / , S/ 'S' / , SS/ 's' ,
     )     O/'O' / , QQ/'q' / , T/'T' / , T?/'t' /
     C
0007      IREF = 0
0008      IDISK = 0
0009      IOR = 0
0010      IOW = 1
     C
0011      WRITE (IOW,1247)
0012 1247  FORMAT (/////,     ----   BIBLIOGRAPHIC PRINTING   ----   .///.
     1          /////,
     1'LAST REVISION  20/01/87  version 4.0'/////,10X, ...c... ////)
     C
0013      CALL OPEN (3,'BIB.PAR')
     C
0014      WRITE (IOW,*) ' USING MULTI-DISK FILES? (Y)es (N)o'
0015      READ (IOR,1226) FILE
     C
0016      WRITE (IOW,*) ' PARAMETER FILE OPEN ON DEFAULT DRIVE (BIB.PAR)'
     C
0017 1252  WRITE (IOW,*) ' PRINTING FROM : (S)orted version file, or
0018      WRITE (IOW,*) '                 : (I)nput data version file
0019      READ (IOR,1226) SORT
     C
```

```
0020        WRITE (IOW,*) 'KEY WORDS IN PRINTOUT/OUTPUT   (Y)es  (N)o
0021        READ (IOR,1226) KEY
      C
0022 1260   IF (SORT.EQ.S .OR. SORT.EQ.SS) THEN
0023            WRITE (IOW,1242)
0024 1242       FORMAT (/' NAME BIBLIOGRAPHIC REFERENCE FILE OR <CR> ./.
                         TO DEFAULT TO 'BIBSRT.DAT' ,Z)
0025        ELSE
0026            WRITE (IOW,1245)
0027 1245       FORMAT (/' NAME BIBLIOGRAPHIC REFERENCE FILE OR <CR> ./.
                         TO DEFAULT TO 'BIBLIO.DAT' ,Z)
0028        ENDIF
      C
0029        READ (IOR,1234) FNAME
0030        IF (FNAME(1).EQ.BLANK) THEN
0031            IER = 0
0032            IF (SORT.EQ.S .OR. SORT.EQ.SS) THEN
0033             -  CALL OPEN (4, BIBSRT.DAT ,IER)
0034            ELSE
0035                CALL OPEN (4, 'BIBLIO.DAT ,IER)
0036            ENDIF
0037            IF (IER .EQ. 0) GOTO 1227
0038        ELSE
0039            IER = 0
0040            CALL OPEN (4,FNAME,IER)
0041            IF (IER .EQ. 0) GOTO 1227
0042        ENDIF
0043            WRITE (IOW,1241)
0044 1241       FORMAT (/,' DATA FILE DOES NOT EXIST  (R)etry   (Q)uit  ,Z)
0045            READ (IOR,1226) EXIST
0046 1226       FORMAT (A1)
0047            IF (EXIST .EQ. Q .OR. EXIST .EQ. QQ) GOTO 999
0048            GOTO 1260
      C
0049 1227   IF (IDISK.EQ.1) GOTO 1225
0050        IDISK = 1
      C
0051        WRITE (IOW,1265)
0052 1265   FORMAT (/  OUTPUT: (86) COLUMN WIDE - STANDARD OUTPUT FORMAT ,
                    /'          (80) COLUMN WIDE - INPUT FORMAT',
                    /'          (50) COLUMN WIDE - NARROW (COLUMN) FORMAT )
0053        READ (IOR,*) WIDE
      C
0054        IF (WIDE.EQ.80 .AND. SORT.NE.S .AND. SORT.NE.SS) THEN
0055           WRITE (IOW,*)   SAME INPUT / OUTPUT - (T)ry again  (C)ontinue
0056           READ (IOR,1226) TRY
0057           IF (TRY .EQ. T .OR. TRY .EQ. TT) THEN
0058               CALL CLOSE (4)
0059               IDISK = 0
0060               GOTO 1262
0061           ENDIF
0062        ENDIF
      C
0063 1228       WRITE (IOW,1246)
0064 1246       FORMAT (//' NAME OUTPUT FILE OR <CR> TO DEFAULT OF './.
                        '<BIBPRT.OUT> ,Z)
      C
0065        READ (IOR,1234) FNAME
0066 1234   FORMAT (10A6)
      C
0067        IER = 0
0068          IF (FNAME(1).EQ.BLANK) THEN
      C
0069                CALL OPEN (2,'BIBPRT.OUT',IER)
0070                IF (IER.EQ.1) THEN
0071                    CALL OPEN (2,'BIBPRT.OUT')
0072                    GOTO 1225
0073                ENDIF
      C
0074          ELSE
```

```
0075            CALL OPEN (2,FNAME,IER)
0076            IF (IER .EQ. 1) THEN
0077                CALL OPEN (2,FNAME)
0078                GOTO 1225
0079            ENDIF
0080          ENDIF
      C
0081            WRITE (IOW,1235)
0082 1235       FORMAT (' FILE ALREADY EXISTS - OK TO OVERWRITE   (Y)es  .
     /                 (N)o   . ?)
0083            READ (IOR,1226) EXIST
0084            IF (EXIST .EQ. Y .OR. EXIST .EQ. YY) GOTO 1225
0085            CALL CLOSE (2)
0086            GOTO 1228
      C
0087 1225  CALL PRINT
      C
0088       WRITE (IOW,*)
0089       WRITE (IOW,*) ' TOTAL NUMBER OF REFERENCES PRINTED :  ,IREF
      C
      C    CHECKING FOR MULTI-DISK FILES
0090       IF (FILE.EQ.Y .OR. FILE.EQ.YY) THEN
0091           CALL CLOSE (4)
      C
0092           WRITE (IOW,*) ' CHANGE DISK .... THEN HIT (R)eady / (Q)uit
0093           READ  (IOR,1226) FILE
      C
0094           IF (FILE.NE.Q .OR. FILE.NE.QQ) GOTO 1260
0095       ENDIF
      C
0096       WRITE (IOW,*)   FILE COMPLETE
      C
0097 999   CALL CLOSE (2)
0098       STOP
0099       END
** Common block BIB    requires  3036 (Decimal), 0BDC (Hex) Bytes
** Common block GET    requires   486 (Decimal), 01E6 (Hex) Bytes
** Generated Code =  2545 (Decimal), 09F1 (Hex) Bytes
```

```
0001 $OPTIONS X,E,L=200,S=200
0002       SUBROUTINE PRINT
      C
      C*************************************************************************
      C*                                                                       *
      C*    SUBROUTINE TO PRINT A BIBLIOGRAPHIC LISTING EITHER AS INPUT OR AS  *
      C*    PREVIOUSLY PREPARED FOR SORTING. THE LATTER IS DONE BY REMOVING    *
      C*    THE SORTING INDEX FIELD AT THE BEGINNING OF EACH RECORD IN THE FILE*
      C*                                                                       *
      C*                PROGRAM BY:    DOUGLAS CLAY                          *
      C*                               MARINE FISHERIES DIVISION             *
      C*                               GULF FISHERIES CENTER                 *
      C*                               DEPARTMENT OF FISHERIES AND OCEANS    *
      C*                               P.O.BOX 5030, MONCTON, N.B., CANADA.  *
      C*                                                                     .
      C*             (version 1 : April    1975 : ICL Fortran IV)            *
      C*    ORIGINAL PROGRAM (BIBLIO)                                        .
      C*             (version 2 : November 1977 : CDC Fortran IV)            *
      C*               VERSION :      3.0       DATE :  25/12/84            *
      C*    RE-WRITTEN FOR 8-BIT MICRO-COMPUTERS                             *
      C*                      :       3.1       DATE :  10/02/86            *
      C*    PROGRAM CONVERTED TO 16 bit MSDOS COMPUTER USING UTAH FORTRAN.   *
      C*************************************************************************
      C
0003       DIMENSION TITLE (50) , AUTHOR (24) , STRING (496) , YEAR (4) ,
     /            HOLD (4)
0004       COMMON /BIB/ STRING, IOW, IOR, IREF, SORT, WIDE, YEAR, KEY
0005       COMMON /SET/ HOLD , TITLE , ORDER , LINENO , AUTHOR , TOTAUT
0006       DATA SPACE /' '/ , ASTER /'*'/ , S/'S'/ , SS/'s'/ , Y/'Y'/ ,
     /      YY/'y'/ , CURLL/'('/
```

```
      C
      C     READING OUT PARAMETERS FROM <BIB.PAR> FOR USE BY PRINTING
      C     .....  AUTSRT are the number of authors in the sort field
      C     .....  NOSRT  are the number of characters from each authors
      C                   name in the sort field
      C     .....  TOTAUT is AUTSRT * NOSRT
      C     .....  TOTSRT is TOTAUT + 4 (year) + 2 (line number or flag and
      C                   reference number)
      C     .....  ORDER  is a character A or C for Author or Chronological
      C                   sort sequence
      C
0007        IF (SORT.EQ.S .OR. SORT.EQ.SS) THEN
0008           READ (3,*,END=998) AUTSRT , NOSRT , TOTAUT , TOTSRT , RECSIZ
0009           READ (3,1,END=998) ORDER
0010 1         FORMAT (A1)
0011        ENDIF
      C
0012        WRITE (IOW,11) IREF
0013 11     FORMAT (' REFERENCE NO. : ',I4,Z)
0014        END = 0
0015        AUTHM1 = AUTSRT - 1
0016        DO 2 I = 1 , 24
0017 2      AUTHOR(I) = SPACE
      C
0018        CALL GETIT (END)
0019             IF (END.EQ.999) RETURN
      C
0020 22     IREF = IREF + 1
      C
0021        WRITE (IOW,23) IREF
0022 23     FORMAT ('\08\\08\\08\\08\',I4,Z)
      C
0023        LOC1 = 0
0024        LINE = 0
0025        NOAUT = 0
      C
0026        DO 24 I = 1 , 4
0027 24     YEAR(I) = HOLD(I)
      C
0028        IF (SORT.EQ.S .OR. SORT.EQ.SS) THEN
      C
      C        BREAKING AUTHOR(S) OUT FROM SORT FIELD
0029           DO 3 IAUT = 1 , AUTSRT
0030           K = ((IAUT-1) * NOSRT) + 1
      C        CHECKING FOR 'BLANK' (LAST) AUTHOR
0031           IF (STRING(K).EQ.SPACE) GOTO 4
0032           LOC = K - 1
      C
0033           DO 26 J = 1 , NOSRT
0034           LOC = LOC + 1
0035 26        AUTHOR(J) = STRING(LOC)
0036           J = NOSRT
      C
0037 27        LOC1 = LOC1 + 1
      C
0038           IF (LOC1.GT.50) THEN
0039               LOC1 = 0
0040               CALL GETIT (END)
0041               IF (END.EQ.999) RETURN
0042               GOTO 27
0043           ENDIF
      C
0044           IF (TITLE(LOC1).NE.ASTER) THEN
0045              J = J + 1
0046              AUTHOR(J) = TITLE(LOC1)
0047              GOTO 27
0048           ELSE
0049              IF (IAUT.EQ.1) THEN
0050                  CALL PUTIT (1)
0051              ELSE
0052                  CALL PUTIT (2)
0053              ENDIF
      C
```

```
0054          ENDIF
    C
0055             DO 3 I = 1 , 24
0056             AUTHOR(I) = SPACE
    C
0057 3        CONTINUE
    C
0058       ELSE
    C
0059          CALL PUTIT (1)
0060          IF (TITLE(1).EQ.SPACE) THEN
0061              GOTO 4
0062          ELSE
0063              CALL PUTIT (3)
0064              GOTO 6
0065          ENDIF
    C
0066       ENDIF
    C
0067 32    J = 0
0068 33    LOC1 = LOC1 + 1
    C
0069       IF (LOC1.GT.50) THEN
0070           LOC1 = 0
0071           CALL GETIT (END)
0072            IF (END.EQ.999) RETURN
0073           GOTO 33
0074       ENDIF
    C
0075       IF (TITLE(LOC1).NE.ASTER) THEN
0076           J = J + 1
0077           AUTHOR(J) = TITLE(LOC1)
0078           GOTO 33
0079       ELSE
0080           IF (J.EQ.0) GOTO 4
0081           CALL PUTIT (2)
    C
0082           DO 35 I = 1 , 24
0083 35        AUTHOR(I) = SPACE
0084           GOTO 32
0085       ENDIF
    C
0086 4     CALL GETIT (END)
0087            IF (END.EQ.999) RETURN
    C
0088       IF (SORT.EQ.S .OR. SORT.EQ.SS) THEN
    C
0089           CALL PUTIT (3)
    C
0090       ELSE
    C
0091          CALL PUTIT (2)
0092          IF (TITLE(1).EQ.SPACE) THEN
0093              GOTO 4
0094          ELSE
0095              CALL PUTIT (3)
0096              GOTO 6
0097          ENDIF
    C
0098       ENDIF
    C
0099 6     CALL GETIT (END)
0100           IF (END.EQ.999) RETURN
    C
0101       IF (SORT.EQ.S .OR. SORT.EQ.SS) THEN
    C
0102          IF (LINENO.EQ.ASTER) THEN
0103              GOTO 22
0104          ELSE
```

```
0105              IF ((KEY .NE. Y .AND. KEY .NE. YY) .AND.
     >                              (TITLE(1) .EQ. CURLL))  GOTO 6
0106              CALL PUTIT (4)
0107              GOTO 6
0108           ENDIF
     C
0109        ELSE
     C
0110           IF (HOLD(1).NE.SPACE) THEN
0111              GOTO 22
0112           ELSE
0113              IF ((KEY .NE. Y .AND. KEY .NE. YY) .AND.
     >                              (TITLE(1) .EQ. CURLL))  GOTO 6
0114              CALL PUTIT (4)
0115              GOTO 6
0116           ENDIF
     C
0117        ENDIF
     C
0118        RETURN
0119 998    WRITE (IOW,*)  ' ERROR : PARAMETER FILE IS EITHER INCOMPLETE
0120        WRITE (IOW,*)  '          OR ON THE WRONG DRIVE'
0121        RETURN
0122        END
** Common block BIB    requires  3036 (Decimal), 0BDC (Hex) Bytes
** Common block GET    requires   486 (Decimal), 01E6 (Hex) Bytes
** Generated Code =  2059 (Decimal), 080B (Hex) Bytes
```

```
0001 $OPTIONS X,E,L=100,S=100
0002        SUBROUTINE PUTIT (TYPE)
     C*************************************************************************
     C*                                                                       *
     C*    SUBROUTINE TO PRINT OUT DATA IN ONE OF 3 FORMATS:-                 *
     C*          1. STANDARD FORMAT  -  86 COLUMNS WIDE                       *
     C*          2. INPUT FORMAT     -  80 COLUMNS WIDE                       *
     C*          3. NARROW FORMAT    -  50 COLUMNS WIDE.                      *
     C*                                                                       *
     C*     THE 4 TYPES REPRESENT:- 1. FIRST AUTHOR, 2. SUBSEQUENT AUTHORS    *
     C*                             3. FIRST TITLE,  4. SUBSEQUENT TITLES     *
     C*                                                                       *
     C*               PROGRAM BY:    DOUGLAS CLAY                           *
     C*                              MARINE FISHERIES DIVISION              *
     C*                              GULF FISHERIES CENTER                  *
     C*                              DEPARTMENT OF FISHERIES AND OCEANS     *
     C*                              P.O.BOX 5030, MONCTON, N.B., CANADA.   *
     C*                                                                       *
     C*               VERSION :      1.0       DATE :  03/02/85             *
     C*    WRITTEN FOR 8-BIT MICRO-COMPUTERS                                  *
     C*                    :        1.1       DATE :  10/03/86             *
     C*    PROGRAM CONVERTED TO 16 bit MSDOS COMPUTER USING UTAH FORTRAN.     *
     C*                    :        2.0       DATE :  20/01/87             *
     C*                                                                       *
     C*    PROGRAM NOW HANDLES ADDITIONAL KEYWORDS ADDED AT THE END OF THE    *
     C*    REFERENCE. THESE ARE STORED IN GROUPS OF UP TO 48 CHARACTERS EACH   *
     C*    AND BOUNDED BY CURLY BRACKETS { ... }.                             *
     C*************************************************************************
     C
0003        DIMENSION  HOLD(4), TITLE(50), STRING(496), AUTHOR(24), YEAR(4)
0004        COMMON /BIB/ STRING, IOW, IOR, IREF, SORT, WIDE, YEAR, KEY
0005        COMMON /GET/ HOLD , TITLE , ORDER , LINENO , AUTHOR , TOTAUT
     C
0006        IOUT = 2
     C
0007        GOTO (1,2,3,4), TYPE
     C
0008 1      IF (WIDE.EQ.86) THEN
0009           WRITE (IOUT,15) IREF , AUTHOR
0010 15        FORMAT ('['I4']' , 24A1 , I)
0011        ELSE
```

```
0012            IF (WIDE.EQ.80) THEN
0013                WRITE (IOUT,16) YEAR , AUTHOR
0014 16             FORMAT (4A1 , 2X , 24A1 , Z)
0015            ELSE
0016                WRITE (IOUT,18) IREF , AUTHOR
0017 18             FORMAT (/, '[',I4,'] ' , 24A1 , Z)
0018            ENDIF
0019        ENDIF
0020        RETURN
      C
0021 2      IF (WIDE.NE.30) THEN
0022            WRITE (IOUT,25) AUTHOR
0023 25         FORMAT (/,6X , 24A1 , Z)
0024        ELSE
0025            WRITE (IOUT,27) AUTHOR
0026 27         FORMAT (/,7X , 24A1 , Z)
0027        ENDIF
0028        RETURN
      C
0029 3      IF (WIDE.EQ.86) THEN
0030            WRITE (IOUT,35) YEAR, TITLE
0031 35         FORMAT ('('4A1')'  , 50A1)
0032        ELSE
0033            IF (WIDE.EQ.80) THEN
0034                WRITE (IOUT,36) TITLE
0035 36             FORMAT (50A1)
0036            ELSE
0037                WRITE (IOUT,37) YEAR, TITLE
0038 37             FORMAT ('('4A1')'  , / , 50A1)
0039            ENDIF
0040        ENDIF
0041        RETURN
      C
0042 4      IF (WIDE.EQ.86) THEN
0043            WRITE (IOUT,45) TITLE
0044 45         FORMAT (36X , 50A1)
0045        ELSE
0046            IF (WIDE.EQ.80) THEN
0047                WRITE (IOUT,46) TITLE
0048 46             FORMAT (30X , 50A1)
0049            ELSE
0050                WRITE (IOUT,47) TITLE
0051 47             FORMAT (50A1)
0052            ENDIF
0053        ENDIF
      C
0054        RETURN
0055        END
** Common block BIB    requires  3036 (Decimal), 0BDC (Hex) Bytes
** Common block GET    requires   486 (Decimal), 01E6 (Hex) Bytes
** Generated Code =    723 (Decimal), 02D3 (Hex) Bytes
```

```
0001 $OPTIONS X,E,L=100,S=100
0002        SUBROUTINE GETIT (END)
     C*****************************************************************************
     C*                                                                           *
     C*    SUBROUTINE TO GET DATA IN EITHER THE SORTED OR THE INPUT FORMAT.       *
     C*                                                                           *
     C*                PROGRAM BY:    DOUGLAS CLAY                              *
     C*                               MARINE FISHERIES DIVISION                   *
     C*                               GULF FISHERIES CENTER                       *
     C*                               DEPARTMENT OF FISHERIES AND OCEANS          *
     C*                               P.O.BOX 5030, MONCTON, N.B., CANADA.        *
     C*                                                                           *
     C*                VERSION :      1.0       DATE :  21/01/85                  *
     C*    WRITTEN FOR 8-BIT MICRO-COMPUTERS                                      *
     C*                      :        1.1       DATE :  10/03/86                  *
     C*    PROGRAM CONVERTED TO 16 bit MSDOS COMPUTER USING UTAH FORTRAN.         *
     C*****************************************************************************
     C
```

```
0003        DIMENSION  HOLD(4), TITLE(50), STRING(496), AUTHOR(24), YEAR(4)
0004        COMMON /BIB/ STRING, IOW, IOR, IREF, SORT, WIDE, YEAR, KEY
0005        COMMON /GET/ HOLD , TITLE , ORDER , LINENO , AUTHOR , TOTAUT
0006        DATA S/'S'/ , SS/ s'/ , A/'A'/ , AA/ a /
      C
0007        IN = 4
      C
0008        IF (SORT.EQ.S .OR. SORT.EQ.SS) THEN
0009            IF (ORDER.EQ.A .OR. ORDER.EQ.AA) THEN
0010               READ (IN,1,END=3) (STRING(I),I=1,TOTAUT), HOLD, JREF,
           )                          LINENO, TITLE
0011 1             FORMAT (200A1)
0012            ELSE
0013               READ (IN,1,END=3) HOLD, (STRING(I),I=1,TOTAUT), JREF,
           )                         LINENO, TITLE
0014            ENDIF
      C
0015        ELSE
0016            READ (IN,2,END=3) HOLD , AUTHOR , TITLE
0017 2          FORMAT (4A1,2X,24A1,50A1)
0018        ENDIF
      C
0019        RETURN
0020 3      END = 999
0021        RETURN
0022        END
** Common block BIB    requires  3036 (Decimal), 0BDC (Hex) Bytes
** Common block GET    requires   486 (Decimal), 01E6 (Hex) Bytes
** Generated Code =   456 (Decimal), 01C8 (Hex) Bytes

       No Compile errors
```

APPENDIX III

Source listing for the program <BIBFMT.FOR>.

```
***** UTAH Fortran 1.0 (Mod 2) ** Compiling File: D:BIBFMT.FOR *****

0001 $OPTIONS L=150,S=150,E,X
     C
     C*************************************************************************
     C*    BIBLIOGRAPHY TRANSFORMATION PROGRAM                                *
     C*                                                                       *
     C*    THIS PROGRAM ALLOWS A BIBLIOGRAPHY FILE TO BE TRANSFORMED FROM THE *
     C*    STANDARD INPUT FORMAT TO EITHER OF TWO DIFFERENT FORMATS, ONE USED *
     C*    FOR MERGING/SORTING OF THE BIBLIOGRAPHY AND THE OTHER USED IN THE  *
     C*    PREPARATION OF THE "KWIC" INDEX FILES READY FOR SORTING.           *
     C*                                                                       *
     C*    THIS PROGRAM IS WRITTEN IN FORTRAN IV OR FORTRAN 66 (CP/M          *
     C*    NEVADA FORTRAN) AND INTERFACES WITH TWO OTHER MODULES IN THE       *
     C*    BIB???.??? PACKAGE. THE SORT IS CONDUCTED BY MICRO-PRO's SUPERSORT.*
     C*                                                                       *
     C*    FOR LARGE BIBLIOGRAPHIES ON SYSTEMS WITHOUT HARD DISKS - MULTI-DISK*
     C*    FILES CAN BE USED.                                                 *
     C*                                                                       *
     C*                PROGRAM BY:    DOUGLAS CLAY                          *
     C*                               MARINE FISHERIES DIVISION             *
     C*                               GULF FISHERIES CENTER                 *
     C*                               DEPARTMENT OF FISHERIES AND OCEANS    *
     C*                               P.O.BOX 5030, MONCTON, N.B., CANADA.  *
     C*                                                                     *
     C*                (version 1 : June 1978 : CDC Fortran IV)             *
     C*     ORIGINAL PROGRAM CALLED <QIKBIB>                                *
     C*                VERSION :      2.0       DATE :  22/12/84           *
     C*     RE-WRITTEN FOR 8-BIT MICRO-COMPUTERS AND EXPANDED TO COMPLETE KWIC*
     C*                        :      2.1       DATE :  10/03/86           *
     C*    PROGRAM CONVERTED TO 16 bit MS-DOS COMPUTERS USING UTAH FORTRAN.   *
     C*                                                                       *
     C*                        :      3.0       DATE :  22/01/87           *
     C*    PROGRAM NOW HANDLES ADDITIONAL KEYWORDS ADDED AT THE END OF THE    *
     C*    REFERENCE. THESE ARE STORED IN GROUPS OF UP TO 48 CHARACTERS EACH  *
     C*    AND BOUNDED BY CURLY BRACKETS { ... }.                             *
     C*************************************************************************
     C
0002       DIMENSION  FNAME (3) , STRING (496) , KEYIDX(7,2) , KEYWRD(2000)
0003       COMMON /BIB/ STRING, IOW, IOR, IREF, LINTOT, IN, IOUT, IDISK
0004       COMMON /QIK/ KEYIDX , KEYWRD , KEYW , ISORT
0005       DATA BLANK /'    '/ , S/'S'/ , SS/'s'/ , Y/'Y'/ , YY/'y'/ ,
          >     Q/'Q'/ , QQ/'q'/
     C
0006       IREF = 0
0007       LINTOT = 0
0008       IDISK = 0
0009       IOR = 0
0010       IOW = 1
0011       IOUT= 2
0012       IN  = 4
     C
0013       WRITE (IOW,1247)
0014 1247  FORMAT (//////,'     ----   BIBLIOGRAPHIC "KWIC"   ---- ',//,
          1                   ----   AND SORT PREPARATION   ---- '////,
          1'LAST REVISION  23/01/87  version 3.0'/////,10X,'...c...'/////)
     C
0015       CALL OPEN (3,'BIB.PAR')
0016       CALL OPEN (5,'BIB.KEY')
     C
0017       WRITE (IOW,*) ' PARAMETER FILE OPEN ON DEFAULT DRIVE <BIB.PAR>'
0018       WRITE (IOW,*) ' KEY WORD FILE OPEN ON DEFAULT DRIVE <BIB.KEY>'
     C
0019       DO 1262 I = 1 , 496
0020 1262  STRING(I) = ' '
     C
0021       WRITE (IOW,*) ' USING MULTI-DISK FILES? (Y)es (N)o '
0022       READ (IOR,1249) FILE
     C
```

```
0023 1260  WRITE (IOW,1242)
0024 1242  FORMAT (/' NAME BIBLIOGRAPHIC REFERENCE FILE OR <CR>',/,
     >             '       TO DEFAULT TO <BIBLIO.DAT> ',Z)
0025       READ (IOR,1234) FNAME
0026       IF (FNAME(1).EQ.BLANK) THEN
0027           IER = 0
0028           CALL OPEN (4,'BIBLIO.DAT',IER)
0029           IF (IER .EQ. 0) GOTO 1227
0030       ELSE
0031           IER = 0
0032           CALL OPEN (4,FNAME,IER)
0033           IF (IER .EQ. 0) GOTO 1227
0034       ENDIF
0035           WRITE (IOW,1241)
0036 1241 .    FORMAT (/,' DATA FILE DOES NOT EXIST   (R)etry   (Q)uit  ,Z)
0037           READ (IOR,1249) K
0038           IF (K .EQ. Q .OR. K .EQ. QQ) GOTO 999
0039           GOTO 1260
     C
0040 1227  IF (IDISK.EQ.1) GOTO 1225
     C
0041       WRITE (IOW,*) ' PREPARING FOR : (S)orting of bibliography file
0042       WRITE (IOW,*) '               : (T)ransforming to "KWIC" index'
0043       WRITE (IOW,*) '               : (I)ntermediate string file'
0044       READ (IOR,1249) ISORT
0045 1249  FORMAT (A1)
     C
0046       WRITE (IOW,*) ' KEY WORDS IN "KWIC" INDEX    (Y)es  (N)o
0047       READ (IOR,1249) KEYW
     C
0048       IF (ISORT .EQ. S .OR. ISORT .EQ. SS) THEN
0049           WRITE (IOW,1246)
0050 1246     FORMAT (//' NAME OUTPUT FILE OR <CR> TO DEFAULT OF ',/,
     >             '<BIBSRT.DAT> ',Z)
0051       ELSE
0052           WRITE (IOW,1248)
0053 1248     FORMAT (//' NAME OUTPUT FILE OR <CR> TO DEFAULT OF ',/,
     >             '<BIBQIK.DAT> ',Z)
0054       ENDIF
     C
0055       READ (IOR,1234) FNAME
0056 1234  FORMAT (10A6)
     C
0057       IER = 0
0058         IF (FNAME(1).EQ.BLANK) THEN
     C
0059           IF (ISORT.EQ.S .OR. ISORT.EQ.SS) THEN
0060               CALL OPEN (2,'BIBSRT.DAT',IER)
0061               IF (IER.EQ.1) THEN
0062                   CALL OPEN (2,'BIBSRT.DAT')
0063                   GOTO 1225
0064               ENDIF
0065           ELSE
0066               CALL OPEN (2,'BIBQIK.DAT',IER)
0067               IF (IER.EQ.1) THEN
0068                   CALL OPEN (2,'BIBQIK.DAT')
0069                   GOTO 1225
0070               ENDIF
0071           ENDIF
     C
0072         ELSE
0073           CALL OPEN (2,FNAME,IER)
0074           IF (IER .EQ. 1) THEN
0075               CALL OPEN (2,FNAME)
0076               GOTO 1225
0077           ENDIF
0078         ENDIF
     C
0079           WRITE (IOW,1235)
0080 1235     FORMAT (' FILE ALREADY EXISTS - OK TO OVERWRITE   (Y)es ',
     >             ' (N)o ', Z)
0081           READ (IOR,1249) K
0082           IF (K .EQ. Y .OR. K .EQ. YY) GOTO 1225
```

```
0083            CALL CLOSE (IOUT)
0084            GOTO 1227
      C
0085 1225   IF (ISORT.EQ.S .OR. ISORT.EQ.3S) THEN
0086            CALL SORT
0087            GOTO 998
0088        ELSE
0089            CALL KWIC
0090            GOTO 998
0091        ENDIF
      C
0092 998    WRITE (IOW,*)
0093        WRITE (IOW,*) ' TOTAL NUMBER OF REFERENCES RE-FORMATTED :  ,IREF
      C
      C     CHECKING FOR MULTI-DISK FILES
0094        IF (FILE.EQ.Y .OR. FILE.EQ.'Y') THEN
0095            CALL CLOSE (IN)
      C
0096            WRITE (IOW,*) ' CHANGE DISK ...' THEN HIT (R)eady  (Q)uit
0097            READ  (IOR,1249) FILE
0098            IDISK = 1
      C
0099            IF (FILE.NE.Q .OR. FILE.NE.QQ) GOTO 1260
0100        ENDIF
      C
0101        WRITE (IOW,*)   FILE COMPLETE
      C
0102 999    CALL CLOSE (IOUT)
0103        CALL CLOSE (3)
0104        STOP
0105        END
** Common block BIB    requires  2018 (Decimal), 07EA (Hex) Bytes
** Common block QIK    requires 12096 (Decimal), 2F40 (Hex) Bytes
** Generated Code =  2509 (Decimal), 09CD (Hex) Bytes
```

```
0001 $OPTIONS X,E,L=250,S=250
0002        SUBROUTINE KWIC
      C
      C*****************************************************************************
      C*                                                                           *
      C*    SUBROUTINE TO PREPARE BIBLIOGRAPHIC LISTING FOR A Key Word In          *
      C*    Context (KWIC) INDEX. THE OUTPUT IS THEN RUN THROUGH A SUBSEQUENT      *
      C*    SUBROUTINE FOR PREPARING THE KWIC INDEX.                               *
      C*                                                                           *
      C*                PROGRAM BY:     DOUGLAS CLAY                               *
      C*                                MARINE FISHERIES DIVISION                  *
      C*                                GULF FISHERIES CENTER                      *
      C*                                DEPARTMENT OF FISHERIES AND OCEANS         *
      C*                                P.O.BOX 5030, MONCTON, N.B., CANADA.       *
      C*                                                                           *
      C*                VERSION :       1.0       DATE :  30/12/84               *
      C*    WRITTEN FOR 8-BIT MICRO-COMPUTERS                                      *
      C*                          :     1.1       DATE :  10/03/86               *
      C*    PROGRAM CONVERTED TO 16 bit MS-DOS COMPUTERS USING UTAH FORTRAN.       *
      C*                                                                           *
      C*                          :     2.0       DATE :  22/01/87               *
      C*  : ROUTINE NOW HANDLES ADDITIONAL KEYWORDS ADDED AT THE END OF THE        *
      C*    REFERENCE. THESE ARE STORED IN GROUPS OF UP TO 48 CHARACTERS EACH      *
      C*    AND BOUNDED BY CURLY BRACKETS { ... }.                                 *
      C*****************************************************************************
      C
0003        DIMENSION TITLE (80) , STRING (496) , KEYIDX(7,2) , KEYWRD(2000)
0004        COMMON /BIB/ STRING, IOW, IOR, IREF, LINTOT, IN, IOUT, IDISK
0005        COMMON /QIK/ KEYIDX , KEYWRD , KEYN , ISORT
0006        DATA  Y /'Y'/ , /Y /'y'/ , BLANK /'       , DOLLAR / $   ,
                  III /'I'/ , II /'I'/ ,
                  XNUM /'0'/ , PERIOD /'.'/ , SPACE /' '/ , PLUS /'+'/
```

```
0007        DATA TITLE /80*' '/ , DASH /'-'/ , COMMA /','/ , COLON /':'/ ,
           >     SEMI /';'/ , CURLL /'('/ , CURLR /')'/
      C
0008        IF (IDISK .EQ. 1) GOTO 1
      C
0009 102    IF (ISORT .NE. III .AND. ISORT .NE. II) THEN
0010            WRITE (IOW,*) ' WIDTH OF KWIC FILE ( <130 ) '
0011            READ (IOR,*) IWIDE
0012            WRITE (IOW,*) ' POSITION OF SORTED KWIC COLUMN ( <80 ) '
0013            READ (IOR,*) ICOL
0014            IF (ICOL.GE.IWIDE) GOTO 102
      C
0015            WRITE (IOW,*) ' REFERENCE NUMBER STARTS AT ..... '
0016            WRITE (IOW,*) '     4 digit integer or <CR> FOR 0 '
0017            WRITE (IOW,*) '     remember leading zero's ie. 10 is 0010'
0018            READ (IOR,103) IR
0019 103        FORMAT (I4)
0020            IF (IR.GT.0) IREF = IR - 1
      C
0021            WRITE (IOW,*) ' READING IN KEY WORD DISCARD LIST ... '
0022            READ (3,*,END=998) ((KEYIDX (I,J) , J=1,2) , I=1,7)
0023            READ (3,107,END=998) (KEYWRD(I) , I=1,2000)
0024 107        FORMAT (A6)
0025            WRITE (IOW,*) ' ... COMPLETED DISCARD LIST'
      C
0026        ELSE
0027            WRITE (IOW,108) IREF
0028 108        FORMAT (' REFERENCE NO. ' , I4 , Z)
0029        ENDIF
      C
0030 1      READ (IN,2,END=999) YEAR , (TITLE(I),I=7,80)
0031 2      FORMAT (A4 , 2X , 74A1)
      C
0032 21     IF (YEAR.NE.BLANK) THEN
      C
      C     INITIALIZING VALUES AT BEGINNING OF EACH REFERENCE
0033            NEWREF = 0
0034            XMARK = SPACE
0035            BEGIN = 0
0036            IREF = IREF + 1
0037            LOC = 1
      C
0038            STRING(LOC)=DOLLAR
0039            LOC = LOC + 1
      C
0040        ELSE
      C
      C         TESTING AUTHOR SPACE FOR ENTRY
0041            IF (TITLE(7).NE.SPACE) THEN
0042                STRING(LOC) = SPACE
0043                LOC = LOC + 1
0044                STRING(LOC) = DOLLAR
0045                LOC = LOC + 1
0046            ELSE
0047                GOTO 4
0048            ENDIF
0049        ENDIF
      C
      C     TESTING AUTHOR SPACE FOR LENGTH OF ENTRY
0050            DO 32 I = 30 , 7 , -1
0051            IF (TITLE(I).NE.SPACE) THEN
0052               DO 3 J = 7 , I
0053               STRING(LOC) = TITLE(J)
0054 3             LOC = LOC + 1
0055               GOTO 4
0056            ENDIF
0057 32         CONTINUE
      C
      C     TESTING TITLE SPACE FOR ENTRY
0058 4          IF (TITLE(31).EQ.SPACE) GOTO 1
      C
0059            IDASH = 0
      C
```

```
0060              IF (BEGIN.EQ.0) THEN
0061                  BEGIN = 1
0062                  STRING(LOC) = SPACE
0063                  LOC = LOC + 1
0064                  STRING(LOC) = XNUM
0065                  LOC = LOC + 1
0066              ENDIF
      C
      C     FINDING LAST CHARACTER IN LINE
0067              DO 5 IHOLD = 80 , 31 , -1
0068 5            IF (TITLE(IHOLD).NE.SPACE) GOTO 6
      C
      C     CHECKING FOR HYPENATED WORDS
0069 6      IF (TITLE(IHOLD).EQ.DASH .AND. TITLE(IHOLD-1).NE.SPACE) THEN
0070            IHOLD = IHOLD - 1
0071            IDASH = 1
0072        ENDIF
      C
      C     PRINTING TITLE UNTIL A PERIOD IS ENCOUNTERED
0073              DO 7 I = 31 , IHOLD
      C
      C     SPACING IMBEDDED PUNCTUATION
      C
0074                  IF (TITLE(I).EQ.COMMA .OR.
     >                    TITLE(I).EQ.COLON .OR.
     >                    TITLE(I).EQ.SEMI  .OR.
     >                    TITLE(I).EQ.DASH )     THEN
      C
0075                      IF (TITLE(I-1).NE.SPACE) THEN
0076                          STRING(LOC) = SPACE
0077                          LOC = LOC + 1
0078                      ENDIF
      C
0079                      STRING(LOC) = TITLE(I)
0080                      LOC = LOC + 1
      C
0081                      IF (TITLE(I+1).NE.SPACE) THEN
0082                          STRING(LOC) = SPACE
0083                          LOC = LOC + 1
0084                      ENDIF
0085                  GOTO 7
0086                  ENDIF
      C
0087              STRING(LOC) = TITLE(I)
0088              LOC = LOC + 1
0089 7            IF (TITLE(I).EQ.PERIOD) GOTO 72
      C
0090              IF (IDASH.EQ.0) THEN
0091                  STRING(LOC) = SPACE
0092                  LOC = LOC + 1
0093              ENDIF
0094              GOTO 1
      C
      C     ADDING KEY WORDS FROM END OF REFERENCE IF THEY ARE REQUIRED
      C
0095 72           IF (KEYW .EQ. Y .OR. KEYW .EQ. YY) THEN
0096                 J = I+1
0097                 DO 73 I = J , IHOLD
0098                 IF (TITLE(I) .EQ. CURLL) THEN
0099                    STRING(LOC) = SPACE
0100                    LOC = LOC+1
0101                    STRING(LOC) = TITLE(I)
0102                    I = I+1
0103                    LOC = LOC+1
0104                    GOTO 74
0105                 ENDIF
0106 73              CONTINUE
0107                 GOTO 76
      C
```

```
     C      READING IN KEYWORDS FROM SUPPLEMENTARY LIST AFTER REFERENCE
0108 74             DO 75 J = I , IHOLD
0109                IF (TITLE(J) .NE. SPACE) THEN
0110                   STRING(LOC) = TITLE(J)
0111                ELSE
0112                   STRING(LOC) = CURLR
0113                   LOC = LOC+1
0114                   STRING(LOC) = SPACE
0115                   LOC = LOC+1
0116                   STRING(LOC) = CURLL
0117                ENDIF
0118                LOC = LOC+1
0119                IF (TITLE(J) .EQ. CURLR) GOTO 76
0120 75             CONTINUE
     C
     C      READING REMAINING REFERENCE TO FIND NEXT KEYWORD LINE
0121 76             READ (IN,2,END=999) YEAR , (TITLE(I),I=7,80)
0122                IF (YEAR.NE.BLANK) THEN
0123                   NEWREF = 1
0124                   GOTO 8
0125                ELSE
     C      FINDING LAST CHARACTER IN LINE
0126                   IF (TITLE(31) .NE. CURLL) GOTO 76
0127                   DO 78 IHOLD = 80 , 31 , -1
0128 78                IF (TITLE(IHOLD).NE.SPACE) GOTO 79
0129 79                I = 30
0130                   GOTO 72
0131                ENDIF
0132             ENDIF
     C
     C      PRINTING OUT REFERENCE IN STRING FORMAT OR KWIC FORMAT
     C       ... STRING FORMAT
     C             A PLUS IN THE FIRST COLUMN INDICATES MORE TO FOLLOW
     C             A SPACE IN THE FIRST COLUMN INDICATES ONLY ONE 248
     C             CHARACTER RECORD
     C
0133 8              IF (ISORT .EQ. III .OR. ISORT .EQ. II) THEN
0134                   IF (LOC.GT.249) XMARK = PLUS
0135                       WRITE (IOUT,81) XMARK , (STRING(I),I=1,248)
0136 81                    FORMAT (249A1)
0137                   IF (LOC.GT.249)
                           WRITE (IOUT,81) (STRING(I),I=249,496)
                           LOC = LOC + 1
0138                   WRITE (IOW,82) IREF
0139 82                FORMAT ('\08\\08\\08\\03\' , I4 , Z)
0140                ELSE
0141       C         ... KWIC FORMAT
0142                   CALL KEY (IWIDE,ICOL,LOC)
0143                ENDIF
     C
     C      CLEARING STRING STORAGE REGISTER
0144            DO 84 I = 1 , LOC
0145 84         STRING(I) = SPACE
0146            LOC = 1
     C
0147       IF (NEWREF .EQ. 0) THEN
     C      READING REMAINING REFERENCE TO FIND NEXT TITLE
0148 96         READ (IN,2,END=999) YEAR , (TITLE(I),I=7,80)
0149            IF (YEAR.EQ.BLANK) GOTO 96
0150       ENDIF
     C
0151       GOTO 21
     C
0152 999   WRITE (IOW,*) ' ERROR *** FILE (BIB.KEY) EITHER MISSING ON'
0153       WRITE (IOW,*) '                DEFAULT DRIVE OR INCOMPLETE'
0154 999   RETURN
0155       END
++ Common block BIB    requires  3018 (Decimal), 0BCA (Hex) Bytes
++ Common block QIK    requires 12096 (Decimal), 2F40 (Hex) Bytes
++ Generated Code =  3384 (Decimal), 0D38 (Hex) Bytes
```

```
0001 $OPTIONS X,E,L=250,S=250
0002       SUBROUTINE KEY (IWIDE,ICOL,TOT)
     C
     C*****************************************************************************
     C*                                                                           *
     C*    SUBROUTINE TO BREAK UP "STRING REFERENCE" INTO KEY WORD LINES.          *
     C*                                                                           *
     C*              PROGRAM BY:         DOUGLAS CLAY                             *
     C*                                  MARINE FISHERIES DIVISION,               *
     C*                                  GULF FISHERIES CENTER                    *
     C*                                  DEPARTMENT OF FISHERIES AND OCEANS,      *
     C*                                  P.O.BOX 5030, MONCTON, NEW BRUNSWICK,    *
     C*                                  CANADA  E1C 9B6.                         *
     C*                                                                           *
     C*              VERSION   :      1.0       DATE :  09/01/85                 *
     C*                        :      1.1       DATE :  10/03/86                 *
     C*    PROGRAM CONVERTED TO 16 bit MS-DOS COMPUTERS USING UTAH FORTRAN.       *
     C*                                                                           *
     C*                        :      2.0       DATE :  22/01/87                 *
     C*    ROUTINE NOW HANDLES ADDITIONAL KEYWORDS ADDED AT THE END OF THE        *
     C*    REFERENCE. THESE ARE STORED IN GROUPS OF UP TO 48 CHARACTERS EACH      *
     C*    AND BOUNDED BY CURLY BRACKETS { ... }.                                 *
     C*****************************************************************************
     C
0003       DIMENSION STRING (496) , KEYIDX (7,2) , KEYWRD (2000) , HOLD(100)
0004       COMMON /BIB/ STRING, IOW, IOR, IREF, LINTOT, IN, IOUT, IDISK
0005       COMMON /QIK/ KEYIDX , KEYWRD , KEYW , ISORT
0006       DATA XNUM /'0'/ , SPACE /' '/ , HOLD /100*' '/ , COMMA / ','/
0007       DATA PERIOD /'.'/ , BLANK /'       '/ , QUOTES/1H'/ , QUOTED/ ' ' / ,
          >     LFTSQU /'['/ , RGTSQU /']'/ , LFTRND / '(' / , RGTRND / ')' / ,
          >     CURLL /'{'/ , CURLR /'}'/
     C
0008       FIRST = 1
0009       LAST = 0
0010       LINE = 0
0011       LOC = 1
0012       KKEY = -1
     C
0013       WRITE (IOW,10) IREF , LINE
0014 10    FORMAT (/' REFERENCE 'I4' LINE 'I2,2)
     C
0015 1     IF (STRING(LOC).NE.XNUM) THEN
     C
0016 2         IF (STRING(LOC).EQ.SPACE) THEN
     C         CALCULATING THE String END MARKER
0017               SEND = FIRST + (IWIDE - ICOL)
     C
0018               IF (FIRST.GE.ICOL) THEN
     C             CALCULATING THE String BEGinning MARKER
0019                   SBEG = FIRST - ICOL
0020                   WRITE (IOUT,3) IREF , (STRING(I), I=SBEG,SEND)
0021 3                 FORMAT (I4,2X,200A1)
0022               ELSE
0023                   IFILL = ICOL - FIRST
0024                   SBEG = 1
0025                   WRITE (IOUT,3) IREF , (HOLD(I),I=1,IFILL) , SPACE ,
          >                               (STRING(I), I=SBEG,SEND)
0026               ENDIF
     C
0027               LINE = LINE + 1
0028               FIRST = LOC + 1
0029               WRITE (IOW,31) LINE
0030 31            FORMAT ('\08\\08\'I2,2)
0031           ENDIF
     C
0032           LOC = LOC + 1
     C
     C         CHECKING FOR EMBEDDED PUNCTUATION ie. BRACKETS or QUOTE'S
0033 33        IF (STRING(FIRST).EQ.LFTRND .OR.
          >        STRING(FIRST).EQ.QUOTES .OR.
          >        STRING(FIRST).EQ.CURLL .OR.
          >        STRING(FIRST).EQ.QUOTED)    THEN
0034               LOC = LOC + 1
```

```
0035                FIRST = LOC
0036                GOTO 33
0037             ENDIF
     C
0038             IF (LOC.GT.TOT) GOTO 8
0039             IF (KKEY) 1,4,4
     C
0040          ELSE
0041             KKEY = 0
0042             STRING(LOC) = SPACE
0043             LOC = LOC + 1
0044             IF (STRING(LOC).EQ.LFTSQU) LOC = LOC + 1
     C
0045 32          FIRST = LOC
0046             IF (STRING(FIRST).EQ.LFTRND .OR.
     >               STRING(FIRST).EQ.CURLL  .OR.
     >               STRING(FIRST).EQ.QUOTES .OR.
     >               STRING(FIRST).EQ.QUOTED)     THEN
0047                LOC = LOC + 1
0048                GOTO 32
0049             ENDIF
     C
0050 4           IF (STRING(LOC).EQ.SPACE) THEN
0051                LAST = LOC - 1
0052                IF (STRING(LAST).EQ.COMMA .OR.
     >                  STRING(LAST).EQ.PERIOD)  LAST = LAST - 1
0053                IF (STRING(LAST).EQ.RGTSQU .OR.
     >                  STRING(LAST).EQ.CURLR  .OR.
     >                  STRING(LAST).EQ.RGTRND)  LAST = LAST - 1
0054                IF (STRING(LAST).EQ.QUOTES .OR.
     >                  STRING(LAST).EQ.QUOTED)  LAST = LAST - 1
     C
0055                NUM = LAST - FIRST + 1
     C
0056                IF (NUM.LE.2) THEN
0057                   KKEY = 1
0058                   GOTO 7
0059                ENDIF
     C
0060                NUM3 = NUM
0061                IF (NUM3.GT.6) NUM3 = 7
     C
     C              START AND END LOCATIONS IN THE KEY WORD FILE
     C               FOR WORDS OF "NUM" SIZE
0062                NUM1 = KEYIDX(NUM3,1)
0063                NUM2 = KEYIDX(NUM3,2)
     C
0064                IF (NUM3.EQ.7) NUM3 = 6
     C
0065                NUM4 = NUM3 + FIRST - 1
     C              BREAKING KEY WORD OUT OF 'STRING' REFERENCE
0066                GOTO ( 51,52,53,54,55,56 ) , NUM3
0067 56                ENCODE (WORD,6,560) (STRING(I),I=FIRST,NUM4)
0068 560               FORMAT (6A1)
0069                   GOTO 5
0070 55                ENCODE (WORD,6,550) (STRING(I),I=FIRST,NUM4)
0071 550               FORMAT (5A1,1X)
0072                   GOTO 5
0073 54                ENCODE (WORD,6,540) (STRING(I),I=FIRST,NUM4)
0074 540               FORMAT (4A1,2X)
0075                   GOTO 5
0076 53                ENCODE (WORD,6,530) (STRING(I),I=FIRST,NUM4)
0077 530               FORMAT (3A1,3X)
0078                   GOTO 5
0079 52                ENCODE (WORD,6,520) (STRING(I),I=FIRST,NUM4)
0080 520               FORMAT (2A1,4X)
0081                   GOTO 5
0082 51                ENCODE (WORD,6,510) (STRING(I),I=FIRST,NUM4)
0083 510               FORMAT (1A1,5X)
     C
0084 5              KKEY = 0
0085                IF (NUM1.LE.0 .OR. NUM2.LE.0) GOTO 7
```

```
0086                DO 6 I = NUM1 , NUM2
0087                IF (KEYWRD(I).EQ.BLANK) GOTO 7
0088                IF (WORD.EQ.KEYWRD(I)) KKEY = 1
0089 6              IF (KKEY.EQ.1) GOTO 7
     C
0090 7              IF (KKEY.EQ.0) GOTO 2
0091                LOC = LOC + 1
0092                IF (LOC.LT.TOT) GOTO 32
0093            ELSE
0094                LOC = LOC + 1
0095                GOTO 4
0096            ENDIF
0097        ENDIF
     C
0098 8      CONTINUE
     C
0099        RETURN
0100        END
** Common block BIB    requires  3018 (Decimal), 0BCA (Hex) Bytes
** Common block QIK    requires 12096 (Decimal), 2F40 (Hex) Bytes
** Generated Code =  2292 (Decimal), 08F4 (Hex) Bytes
```

```
0001 $OPTIONS X,E,L=230,S=230
0002        SUBROUTINE SORT
     C
     C*************************************************************************
     C*                                                                       *
     C*    SUBROUTINE TO PREPARE BIBLIOGRAPHIC LISTING FOR SORTING BY ADDING  *
     C*    A SORTING INDEX FIELD AT THE BEGINNING OF EACH RECORD IN THE FILE. *
     C*                                                                       *
     C*                PROGRAM BY:    DOUGLAS CLAY                          *
     C*                               MARINE FISHERIES DIVISION             *
     C*                               GULF FISHERIES CENTER                 *
     C*                               DEPARTMENT OF FISHERIES AND OCEANS     *
     C*                               P.O.BOX 5030, MONCTON, N.B., CANADA.  *
     C*                                                                     *
     C*             (version 1 : April    1975 : ICL Fortran IV)            *
     C*    ORIGINALLY A SUBROUTINE OF THE PROGRAM (PROBIB)                  *
     C*             (version 2 : November 1977 : CDC Fortran IV)            *
     C*                VERSION :      3.0       DATE :  26/12/84           *
     C*    RE-WRITTEN FOR 8-BIT MICRO-COMPUTERS                             *
     C*                       :      3.1       DATE :  10/03/86           *
     C*    PROGRAM CONVERTED TO 16 bit MS-DOS COMPUTERS USING UTAH FORTRAN.  *
     C*************************************************************************
     C
0003        DIMENSION TITLE (50) , AUTHOR (24) , STRING (496) , YEAR (4) ,
           >          HOLD (4) , CHAR (26)
0004        COMMON /BIB/ STRING, IOW, IOR, IREF, LINTOT, IN, IOUT, IDISK
0005        DATA SPACE /' '/ , ASTER /'*'/ , A/'A'/ , AA/'a' / , PLUS/'+'/
0006        DATA  CHAR /'A','B','C','D','E','F','G','H',
           >            'I','J','K','L','M','N','O','P',
           >            'Q','R','S','T','U','V','W','X','Y','Z'/
     C
0007        JREF = 1
     C
0008        IF (IDISK.EQ.1) THEN
0009            IREF = IREF + 1
0010            WRITE (IOW,10) IREF
0011 10         FORMAT (' REFERENCE NO. ',I4,Z)
0012            GOTO 11
0013        ENDIF
     C
     C      NUMBER OF AUTHORS TO INCLUDE IN SORT ID HEADER - TRY 3
0014        WRITE (IOW,*) ' SORT INDEX FIELD MADE UP OF '
0015        WRITE (IOW,*) '       NUMBER OF AUTHORS ( <3 )'
0016        READ (IOR,*) AUTSRT
```

```
      C     NUMBER OF CHARACTERS FROM EACH AUTHORS NAME TO SORT ON - TRY 7
0017        WRITE (IOW,*)'       CHARACTERS FROM EACH AUTHOR FOR SORT ( <15 )'
0018        READ (IOR,*) NOSRT
0019        WRITE (IOW,*)' SORT (C)HRONOLOGICALLY, THEN BY AUTHOR; OR BY'
0020        WRITE (IOW,*)'       (A)UTHOR, THEN CHRONOLOGICALLY.
0021        READ (IOR,2) ORDER
      C
0022        AUTHM1 = AUTSRT - 1
0023        TOTAUT = AUTSRT * NOSRT
0024        TOTSRT = AUTSRT * NOSRT + 4 + 2
0025        RECSIZ = TOTSRT + 50
      C
      C     WRITING OUT PARAMETERS ON <BIB.PAR> FOR USE BY PRINTING
0026        WRITE (3,*) AUTSRT , ',' , NOSRT , ',' , TOTAUT ,
     )                  TOTSRT , ',' , RECSIZ
0027        WRITE (3,2) ORDER
      C
0028        IREF = IREF + 1
      C
0029        WRITE (IOW,10) IREF
      C
0030        HOLD(1) = SPACE
0031        LINE =  0
0032        LOC  =  0
0033        LOC1 =  199
0034        NOAUT = 0
0035 11     READ (IN,2,END=999) YEAR , AUTHOR , TITLE
0036 2      FORMAT (4A1,2X,24A1,50A1)
      C
0037 21     IF (HOLD(1).EQ.SPACE) THEN
0038            DO 22 I = 1 , 4
0039 22        HOLD(I) = YEAR(I)
0040        ENDIF
      C
      C     NUMBER OF CHARACTERS IN AUTHORS NAME FIELD
0041            NOCHAR = 0
0042            DO 23 I = 24 , 1 , -1
0043              IF (AUTHOR(I).NE.SPACE) THEN
0044                  NOCHAR = I
0045                  GOTO 25
0046              ENDIF
0047 23           CONTINUE
      C
      C     BREAKING AUTHOR(S) INTO SORT FIELD
0048 25       NOAUT = NOAUT + 1
0049              DO 3 I = 1 , NOSRT
0050              LOC = LOC + 1
0051 3            STRING (LOC) = AUTHOR (I)
0052          IF (NOSRT.GE.NOCHAR) THEN
      C       THE AUTHORS NAME IS COMPLETELY WITHIN SORT FIELD - FLAG IT
      C       WITH AN ASTERISK
0053              LOC1 = LOC1 + 1
0054              STRING (LOC1) = ASTER
0055              GOTO 33
0056          ENDIF
      C
      C       ADDING REMAINDER OF AUTHORS NAME TO HOLDING STRING -
      C       WHERE ALL OF NAME DID NOT FIT IN SORT FIELD
0057          NOSRT1 = NOSRT + 1
0058              DO 31 I = NOSRT1 , 24
0059              LOC1 = LOC1 + 1
0060              STRING (LOC1) = AUTHOR (I)
0061 31           IF (I.GE.NOCHAR) GOTO 32
      C
0062 32       LOC1 = LOC1 + 1
0063          STRING (LOC1) = ASTER
      C
0064 33       IF (TITLE(1).EQ.SPACE) THEN
0065              IF (NOAUT.LE.AUTHM1) GOTO 11
0066          ELSE
0067              GOTO 71
0068          ENDIF
      C
```

```
0069 5     READ (IN,6,END=999) AUTHOR , TITLE
0070 6     FORMAT (6X , 24A1 , 50A1)
     C
     C     NUMBER OF CHARACTERS IN AUTHORS NAME FIELD
0071           NOCHAR = 0
0072           DO 61 I = 24 , 1 , -1
0073             IF (AUTHOR(I).NE.SPACE) THEN
0074                 NOCHAR = I
0075                 GOTO 62
0076             ENDIF
0077 61         CONTINUE
     C
0078 62         DO 7 I = 1 , NOCHAR
0079            LOC1 = LOC1 + 1
0080 7          STRING (LOC1) = AUTHOR (I)
     C
0081             LOC1 = LOC1 + 1
0082             STRING (LOC1) = ASTER
     C
0083            IF (TITLE(1).EQ.SPACE) GOTO 5
     C
0084 71         LOC1 = LOC1 + 1
0085            STRING(LOC1) = ASTER
0086            LINE = LINE + 1
     C
     C     WRITING OUT SORT FIELD INCLUDING LINE NUMBER OF INDIVIDUAL REFERENCE
     C     BEFORE EACH ACTUAL LINE OF REFERENCE
     C     FIRST ONE OR TWO TITLE LINES CONTAIN THE REMAINS OF THE AUTHORS
     C     NAMES AS REQUIRED - THEY ARE FLAGGED BY ASTERISKS IN THE LINE

     C     NUMBER COLUMN - ALSO BY AN ASTERISK SEPARATING EACH AUTHOR AND
     C     ONE AT THE END
     C
0087         IF (LOC1.GT.199) THEN
0088             IF (ORDER.EQ.A .OR. ORDER.EQ.AA) THEN
0089                WRITE (IOUT,85) (STRING(I),I=1,TOTAUT), HOLD, CHAR(JREF),
            >                        ASTER, (STRING(I),I=200,249)
0090             ELSE
0091                WRITE (IOUT,85) HOLD, (STRING(I),I=1,TOTAUT), CHAR(JPEF),
            >                        ASTER, (STRING(I),I=200,249)
0092             ENDIF
     C
0093             DO 82 I = 200 , 249
0094 82          STRING(I) = SPACE
     C
0095             IF (LOC1.GT.249) THEN
0096                 IF (ORDER.EQ.A .OR. ORDER.EQ.AA) THEN
0097                    WRITE (IOUT,85) (STRING(I),I=1,TOTAUT), HOLD,
            >                  CHAR(JREF), PLUS, (STRING(I),I=250,299)
0098                 ELSE
0099                    WRITE (IOUT,85) HOLD, (STRING(I),I=1,TOTAUT),
            >                  CHAR(JREF), PLUS, (STRING(I),I=250,299)
0100                 ENDIF
     C
0101                 DO 83 I = 250 , 299
0102 83             STRING(I) = SPACE
0103             ENDIF
0104         ENDIF
     C
0105 84          IF (ORDER.EQ.A .OR. ORDER.EQ.AA) THEN
0106                WRITE (IOUT,85) (STRING(I),I=1,TOTAUT), HOLD, CHAR(JREF),
            :>                       CHAR(LINE), TITLE
0107 85             FORMAT (200A1)
0108             ELSE
0109                WRITE (IOUT,85) HOLD, (STRING(I),I=1,TOTAUT), CHAR(JREF),
            >                        CHAR(LINE), TITLE
0110             ENDIF
     C
0111             READ (IN,2,END=999)  YEAR , AUTHOR , TITLE
0112             LINE = LINE + 1
0113             IF (YEAR(1).EQ.SPACE) GOTO 84
0114                 DO 86 I = 1 , TOTAUT
0115 86             STRING(I) = SPACE
```

```
      C
0116                  IREF = IREF + 1
0117                  IF (JREF.LT.25) THEN
0118                     JREF = JREF + 1
0119                  ELSE
0120                     JREF = 0
0121                  ENDIF
0122                  WRITE (IOW,88) IREF
0123 88               FORMAT ('\08\\08\\08\\08\',I4,Z)
      C
0124                HOLD(1) = SPACE
0125                IF (LINE.GE.15) LINE = 0
0126                LOC  =  0
0127                LOC1 =  199
0128                NOAUT = 0
0129             GOTO 21
      C
0130 999    CONTINUE
      C     BLANKING AUTHOR IN CASE OF MULTI-DISK FILES
0131         HOLD(1) = SPACE
0132                IF (LINE.GE.15) LINE = 0
0133                LOC  =  0
0134                LOC1 =  199
0135                NOAUT = 0
0136        RETURN
0137        END
** Common block BIB    requires  3018 (Decimal), 0BCA (Hex) Bytes
** Generated Code =  2949 (Decimal), 0B85 (Hex) Bytes


       No Compile errors
```